

CRWR Online Report 05-1

**Map to Map: Converting a NEXRAD Rainfall Map into a Flood
Inundation Map**

by

Oscar Robayo, B. S.; M.Sc.

Graduate Research Assistant

and

David R. Maidment, PhD

Principal Investigator

January 2005

CENTER FOR RESEARCH IN WATER RESOURCES

The University of Texas at Austin

J.J. Pickle Research Campus

Austin, TX 78712-4497

This document is available online via World Wide Web at

<http://www.crwr.utexas.edu/online.shtml>

Acknowledgements

The authors would also like to thank the San Antonio River Authority, the City of San Antonio, the Bexar County, and PBS&J Austin for funding and supporting this research project.

Table of Contents

List of Figures	vi
List of Tables	x
Chapter 1 Introduction	1
1.1 Background	1
1.2 Motivation	2
1.3 Scope and Objectives	5
Chapter 2 Literature Review	8
2.1 Coupling Methods	8
2.2 GIS-based Floodplain Management	13
2.3 Flood Mitigation Plan Evaluation and Sizing	19
2.4 Proposed Contributions to Knowledge	21
Chapter 3 Methodology	24
3.1 Conceptual Definitions	24
3.1.1 Identifying Modeling Elements	26
3.1.2 Geographic Representations	26
3.1.3 Flow Change Points	27
3.1.4 Arc Hydro Time Series Extension	27
3.1.5 Time Series Retrieval from HEC-DSS	28
3.1.6 Automated Workflow	28
3.2 Case Study: The San Antonio River Basin	28
3.3 Input Datasets	30
3.3.1 Precipitation Datasets	30
3.3.2 Hydrologic and Hydraulic Models	33
3.3.3 Geographic Representations in Arc Hydro	34
3.3.4 Terrain Surface Dataset	37

3.4 Floodplain Modeling	38
3.4.1 Transferring Time Series to HEC-DSS	39
3.4.2 External Execution of HEC-HMS	40
3.4.3 Transferring Flow Records to Geodatabase	40
3.4.4 External Execution of HEC-RAS	41
3.4.5 Automated Floodplain Delineation	41
3.5 Integrating Elements	41
3.5.1 The Hosting Geographic Environment	42
3.5.2 Mapping Time Series to Watersheds	44
3.5.3 Model-Specific Elements	45
3.5.4 Model Codification in Arc Hydro	48
3.5.5 Geographic Association of Exchange Points	51
3.5.6 Model Connectivity in Arc Hydro	54
3.5.7 Time Series Transfer to Arc Hydro	57
3.5.8 Updating the Hydraulic Model Configuration	61
3.5.9 Automated Floodplain Delineation	66
Chapter 4 Map2Map Components	69
4.1 Modeling Components for the Historic Use Case.....	69
4.1.1 The Radar2GDB Script Tool.....	70
4.1.2 The Time Series Transfer Script Tool.....	75
4.1.3 The GDB2HMSDSS Script Tool	79
4.1.4 The HEC-HMS Caller Script Tool.....	102
4.1.5 The DSS to GDB Script Tool.....	109
4.1.6 The Geographic Component	112
4.1.7 Time Series Extraction	121
4.1.8 HEC-DSS and Arc Hydro Associations.....	125
4.1.9 The GDB to RASDSS Script Tool	138
4.1.10 The HEC-RAS Caller Script Tool.....	145
4.1.11 The SDF2XML Script Tool	154

4.1.12 The XML2GDB Script Tool	158
4.1.13 Spatial Manipulations of Hydraulic Modeling Results	162
4.2 Model Components for the Design Storm Use Case.....	177
4.2.1 The DesignStormSeries Model	181
4.2.2 The DigitalStorm2HMS Script Tool	183
Chapter 5 Conclusions and Recommendations	186
5.1 A Seamless End-to-End Geographic Integration	186
5.2 Model Code Elements as Geographic Integrators	188
5.3 Rainfall Inputs of various Forms.....	190
5.4 Exchange of Spatially-Referenced Time Series	192
5.5 An Ensemble of Models and Tools	194
5.6 A True Spatial Approach.....	195
5.7 A Regional Approach.....	195
5.8 Contributions to Technology.....	196
5.9 Limitations	197
5.10 Follow-on Research.....	200
Appendix A	202
Appendix B	215
References	218
Vita	224

List of Figures

Figure 1.1 Spatial Integration through GIS Data Models	6
Figure 2.1 Integration of GIS and Environmental Models.....	9
Figure 3.1 Floodplain Modeling Schematic	25
Figure 3.2 Conceptual Elements to Convert a Rainfall Map to a Floodplain Map	25
Figure 3.3 Relative Location of Salado and Rosillo Creeks in Texas.....	30
Figure 3.4 Radar data in ASCII format	32
Figure 3.5 NEXRAD square cells (873 cells) covering the San Antonio River Basin, Texas	33
Figure 3.6 Geographic Representation (b) of Hydrologic Model (a).....	36
Figure 3.7 Geographic Representation (b) of Hydraulic Model (a).....	37
Figure 3.8 Time Series Transfer Component	40
Figure 3.9 Simulation Models Integrated Through Arc Hydro.....	42
Figure 3.10 SchematicNetwork for Salado and Rosillo Creeks.....	46
Figure 3.11 Geographic and Hydrologic Correspondence.....	47
Figure 3.12 Geographic and Hydraulic Correspondence	48
Figure 3.13 Points of Information Exchange in HEC-HMS	50
Figure 3.14 Points of Information Exchange in HMS and RAS	51
Figure 3.15 Model Connectivity through Flow Change Locations.....	52
Figure 3.16 Geographic Representation of Hydrologic Elements	53
Figure 3.17 Conceptual Approach for Model Connectivity in Arc Hydro ..	55
Figure 3.18 HydroJuntion Features as Connectors with Hydrology and Hydraulics	56
Figure 3.19 Relationship Classes for Integration Support	57
Figure 3.20 DSS Time Series content of HMS for Exchange Points.....	59
Figure 3.21 DSSTSCatalog in Arc Hydro.....	60
Figure 3.22 Extended Time Series Component in Arc Hydro	61
Figure 3.23 Creating a Flow File from geodatabase content	63

Figure 3.24 Components of a Steady State Flow File.....	64
Figure 3.25 Time Series Transfer Schematic	65
Figure 3.26 Floodplain Delineation Schematic.....	68
Figure 4.1 Map2Map Workflow in ModelBuilder for Historic Event Use Case	70
Figure 4.2 List of Date Filenames defining a Simulation Time Window	72
Figure 4.3 Time Series values associated with NEXRAD cells in Arc Hydro.....	74
Figure 4.4 Radar2GDB Model Component	75
Figure 4.5 Subbasin Partition based on NEXRAD Cell Overlaps	76
Figure 4.6 Associating NEXRAD Data with Watersheds.....	77
Figure 4.7 TSTransfer Model Component	78
Figure 4.8 Time Series Tables and Related Subbasins	79
Figure 4.9 2-D Array storing watershed's counter and HEC-DSS descriptors	92
Figure 4.10 GDB2HMSDSS Model Component.....	101
Figure 4.11 Time Series Transfer between Arc Hydro and HEC-DSS.....	102
Figure 4.12 HMSCaller Model Component.....	103
Figure 4.13 DSS2GDB Model Component.....	110
Figure 4.14 Arc Hydro and HMS connectivity through SchematicNode Relationship.....	114
Figure 4.15 Relationship Class for Arc Hydro and HEC-HMS connection	115
Figure 4.16 ModelCodes Populated under the SchematicNode Topology	116
Figure 4.17 DSS Time Series Catalog in Arc Hydro	117
Figure 4.18 The Arc Hydro TimeSeries Table.....	118
Figure 4.19 The Arc Hydro TSType Table	118
Figure 4.20 The Midnight Duality	124
Figure 4.21 Filtering HEC-DSS records through Arc Hydro.....	130
Figure 4.22 D part in Arc Hydro's DSSCatalog	131
Figure 4.23 E part in Arc Hydro's DSSCatalog.....	132

Figure 4.24 F part in Arc Hydro's DSSCatalog	134
Figure 4.25 Unit Definition in Arc Hydro's DSSCatalog	134
Figure 4.26 GDB2RASDSS Model Component.....	138
Figure 4.27 Geometry and Flow Combinations for Plan Configurations in RAS	144
Figure 4.28 RASCaller ModelBuilder Component.....	146
Figure 4.29 HEC-RAS Object Library Reference in Visual Basic	147
Figure 4.30 HEC-RAS Object Library and Components.....	148
Figure 4.31 ProjectName Property in RAS Library	149
Figure 4.32 SetComputeWindowToNotShow Method in RAS Library	150
Figure 4.33 RunSteady Method in RAS Library.....	151
Figure 4.34 ExportGIS Method in RAS Library	153
Figure 4.35 Quit Method in RAS Library	154
Figure 4.36 XML File with Tag Definitions for Water Resources	156
Figure 4.37 SDF2XML Model Component	158
Figure 4.38 HEC-RAS Output in XML Content.....	158
Figure 4.39 HEC-RAS Output in Arc Hydro	159
Figure 4.40 XML2XSElev Model Component	160
Figure 4.41 Terrain Model of Rosillo Creek Represented by 5 Foot- Contours	164
Figure 4.42 2 Foot-Contour Lines for Rosillo Creek's Floodplain.....	165
Figure 4.43 Rosillo Creek's terrain model as TIN (a) and GRID (b) formats.....	167
Figure 4.44 Flood Map Created by Subtracting Land Surface from Water Surface.....	168
Figure 4.45 Create and Edit TIN Model Component.....	169
Figure 4.46 TIN to Raster Model Component	170
Figure 4.47 Extract by Mask Geoprocess Component.....	171
Figure 4.48 Continuous Water Elevation Surface from Elevations at Cross Sections	172
Figure 4.49 Create Flood Polygon Model Component	173

Figure 4.50 Create Flood Inundation Map Model Workflow (Option 1) ..	174
Figure 4.51 Create Flood Inundation Map Model Workflow (Option 2) ..	176
Figure 4.52 Map2Map Workflow in ModelBuilder for Design Storm Use Case	178
Figure 4.53 Digital Design Rainfall Maps for Texas (color scale values in inches)	180
Figure 4.54 Depth-Duration-Frequency Table in Arc Hydro	182
Figure 4.55 Workflow for Extracting Rain Depths from Digital Design Storms.....	183
Figure 4.56 Section of Meteorologic file for Frequency Storm Method ...	184
Figure 4.57 ModelBuilder Schematic for DesignStorm2HMS script tool.	185

List of Tables

Table 4.1: Date and Time Formats in Windows and HEC-DSS	123
Table 4.2: HEC-DSS Recommended Variable Type Notation	126
Table 4.3: Arc Hydro Recommended Data Parameter Notation.....	127
Table 4.4: Variable Type Equivalence between HEC-DSS and Arc Hydro	129
Table 4.5: Time Interval Equivalence between HEC-DSS and Arc Hydro	133
Table 4.6: Unit Notation Equivalence between HEC-DSS and Arc Hydro.....	135
Table 4.7: Data Type Equivalence between HEC-DSS and Arc Hydro	136
Table 4.8: Time Series Type Equivalence between HEC-DSS and Arc Hydro .	137
Table 4.9: HEC-DSS Descriptors for Data Exchange.....	140

Chapter 1 Introduction

1.1 BACKGROUND

The Flood Control Act of 1936 gave the U.S. Army Corps of Engineers the responsibility to develop flood attenuation structural measures seen at first as the best way to cope with flood hazards (Grigg, 1996). After the 1960s the emphasis started to shift from structural to non-structural measures or at least to realize the relevance of those strategies as essential components of a more comprehensive approach for floodplain management.

Non-structural measures like zoning, flood insurance, relocation, and flood forecasting, warning, and response systems are intended for land use management in the floodplain to prevent and/or properly face potential damage. In general, after a structural flood control plan is in place, complete control of flood water or prevention of all damages is normally not feasible. There is always a residual flood damage risk that remains and which might be addressed by non-structural measures. The development of Flood Hazard Maps for zoning and insurance programs is part of these non-structural components. The first product that needs to be obtained for these measures is a flood inundation map depicting the extent of the inundated areas for a given real or synthetic storm event.

The evaluation of flood damage reduction plans and the definition of non-structural measures require the consideration of hydrologic, hydraulic, and economic components. Watershed hydrology is usually modeled using the HEC-HMS (USACE, 2000a) hydrologic modeling system (for single event simulations)

and River Hydraulics is normally modeled with the HEC-RAS (USACE, 1997) river analysis system (for simulation of one-dimensional and usually steady flow conditions). Each of these independent simulation models is normally executed in a separate and non-integrated fashion and manually exchanging information between them in a loose, tedious, error-prone, and non-systematic fashion. In general, the flow of information between data acquisition systems like radar data, hydrologic and hydraulic modeling systems, and flood mapping utilities are individually and independently accomplished.

To straighten out the traditional step-wise approach this research deals with the development a Flood Simulation System on top of the Arc Hydro Data Model that will streamline the simulation process, manipulation and generation of modeling outputs needed for flood inundation mapping in flood assessment studies and mitigation plan evaluation. That is, a seamless, end-to-end integrated Regional Watershed Modeling System for flood information support termed the “Map2Map” application.

1.2 MOTIVATION

The need to update flood inundation maps and to obtain realistic flood discharge hydrographs in response to specific storm events is clear. In particular, the herein developed Map2Map integrated approach should allow for straight forward evaluations to properly take into account the following modeling issues and limitations:

- Ability to consider Land-use changes that could alter the response of a watershed to rainfall, since flood discharges determined with outdated data

for undeveloped or natural conditions may not reflect discharges expected with developed conditions.

- Overcome that fact that few streams are gauged, and those that are, do not have a record long enough for accurate flood impact analysis and for statistical models to be fitted accurately in flood frequency analysis studies (Haan, 1977, Chow et al., 1988, Bedient et al., 1992).
- Provide complete flood related information beyond the statistical-analysis approach (e.g., USGS StreamStats) that does not provide information about runoff volume and timing.

Rapid evaluation of flood events provides for a clear visualization of areas in need of flood protection or the need for new floodplain policies and regulations. In some cases, “after-the-fact” projects are normally proposed based on the most recent history of flood damages. An end-to-end, integrated, and dynamic system will help the evaluation of a whole range of potential events (Wurbs, et al., 2001) to better identify flood abatement measures.

Floodplain boundaries used for management and regulating purposes are usually outdated. The shape and water elevations of the floodplain have a dynamic character governed by multiple regional and local factors. This dynamic character of the watersheds poses a constant need for remapping the floodplain boundaries due to:

- New planning criteria defined by updated storm frequency analysis that provides more realistic design storms obtained from longer records of data and the inclusion of new extreme events.

- New regional watershed land use due to recent wildfire events, urban sprawl, and agricultural expansion.
- New local floodplain configurations due to recent stream channel modifications, bridges, culverts, weirs, dams, dikes, levees, storm water systems, and floodplain encroachments.

Emerging technologies are transforming the way water management studies are developed. Advances in object-orientation (OO), the Component Object Model, COM-compliant standards, and Relational Database Modeling Systems (RDBMS) have dramatically increased the functionality, compatibility, and power of GIS. These advances have opened up new avenues for integrating computational models with geographic data (Whiteaker and Maidment, 2001). By means of these new technologies it is possible to expand the traditional role of GIS as just a pre- and post-processor utility for specific modeling systems to configure the long awaited “seamless” link between external applications that can be centrally handled inside a GIS spatial framework.

Flood control should be seen as part of integrated strategies of floodplain management and not just as a single-purpose water project goal. Regularly, damage assessment projects considered only the direct benefits of a single flood project, however, additional benefits or even damages on hydraulically and economically interrelated floodplains by a given flood control project could be substantial on the regional level (Olsen et al., 1998).

All the previous statements favor the configuration of an end-to-end integrated and dynamic system that enables “on the fly” simulations to empower operations not only under planning but also under forecasting modes.

1.3 SCOPE AND OBJECTIVES

The main purpose of this research is to develop, using the new innovations in technology, a Flood Simulation System on top of the Arc Hydro Data Model that will streamline the simulation process, manipulation and generation of modeling outputs needed in flood assessment and mitigation plan evaluation. That is, an integrated Regional Watershed Modeling System for flood information support.

The proposed Flood Simulation System relies on three types of models: Process Models, Data Models, and a Workflow Model. *Process models* represent physical or analytical processes, and include the hydrologic model HEC-HMS for rainfall-runoff transformations and the hydraulic model HEC-RAS for channel streamflow routing. *Data models* represent a geographic framework for storing spatial and non-spatial data for a given purpose such water resources applications using the Arc Hydro data model. Data models may also be used for interfacing several process models like is done in this research for integrating HEC-HMS with HEC-RAS. Finally, a *workflow data model* serves as the gluing framework for interconnecting the process and data models. The simulation of the system is normally achieved by the combination of both state conditions of the system and alterations on those state conditions by process models (Reitsma et al., 1994).

Figure 1.1 depicts the concept of this model integration in which the GIS framework provides the workflow model that allows connectivity between a raw geodatabase formatted with the Arc Hydro schema and associated interface data models that allow communication with engineering models outside the GIS realm like HEC-HMS, HEC-RAS, and potentially flood impact analysis systems.

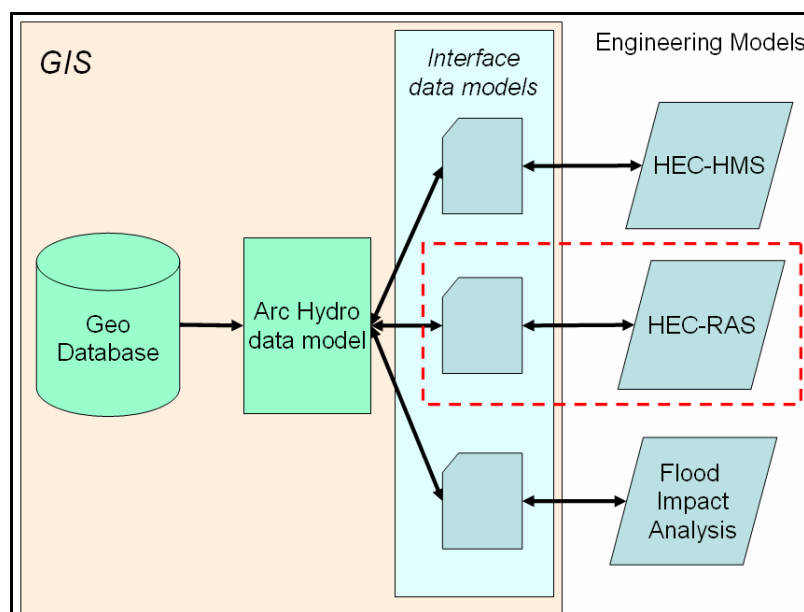


Figure 1.1 Spatial Integration through GIS Data Models

The map-centric framework implements new relational database management file systems and object oriented developments as the central foundation of the GIS-based floodplain modeling system.

This dissertation describes in detail the main phases used in developing a framework for a Geographically Integrated Flood Information System component. The system employs a conjunction of elements from geography, computer

science, and engineering models as key pieces to achieve model integration. The system include elements like a common geographic framework to support the integration and to accommodate specific GIS-based model needs and elements, and the ModelBuilder system in ArcGIS 9 to automate the connection process between supporting tasks and engineering models. The specific objectives of this research project are listed below.

- Develop an end-to-end geographically integrated flood modeling system using ArcGIS and the HEC models, HEC-RAS and HEC-HMS, for the automation of flood inundation maps.
- Generate all the needed components and interfaces to drive the integrated framework with historical and real-time precipitation data from radar (NEXRAD) to generate single event floodplain maps.
- Generate all the needed components and interfaces to drive the integrated framework with synthetic digital precipitation maps to generate probabilistic floodplain maps.
- Implement the proposed modeling integration on a case study for the Salado and Rosillo Creeks in San Antonio, Texas to configure a pilot project and to serve as a proof of concept.

Chapter 2 Literature Review

2.1 COUPLING METHODS

Given the spatial nature of floodplain management components, a GIS-based flood information system is desirable. Coupling methods for integrating GIS and engineering models have been explored since late 1980s as part of the GIS community's efforts to improve the analytical capabilities of GIS (Sui and Maggio, 1999). In spite this effort, it has been stated that GIS is limited in its ability to perform any kind of engineering modeling (Yang and Tsai, 2000) and can only provide for data storage, management, inventory, and mapping functionalities. In flood studies, GIS has normally been used to display the resulting flood boundaries under different formats like vector, raster, and TIN (Azagra et al., 1999). The static "nature" of GIS has been recognized as a large constraint through strong statements in the literature.

"Until GIS has explicit time variation in its data structures, its role will largely be limited to an input data provider, output display, and mapping device" (Maidment, 1993).

Thus, for many years it has been believed that GIS can only contribute to environmental modeling by adding the benefits of its capabilities for handling and storing massive spatially distributed data which is then given a format (via Export Utilities) for the input of a given model or imported after a model simulation is executed for visualization and spatial analysis. In other words, a pure pre- and post-processor functionality has been attributed to GIS.

Different approaches have been used so far to integrate GIS with hydrologic modeling. In general, these approaches can be grouped into four categories: Embedding GIS in hydrological modeling, embedding hydrological modeling in GIS, loose coupling, and tight coupling. Each coupling approach is conceptually shown in Figure 2.1.

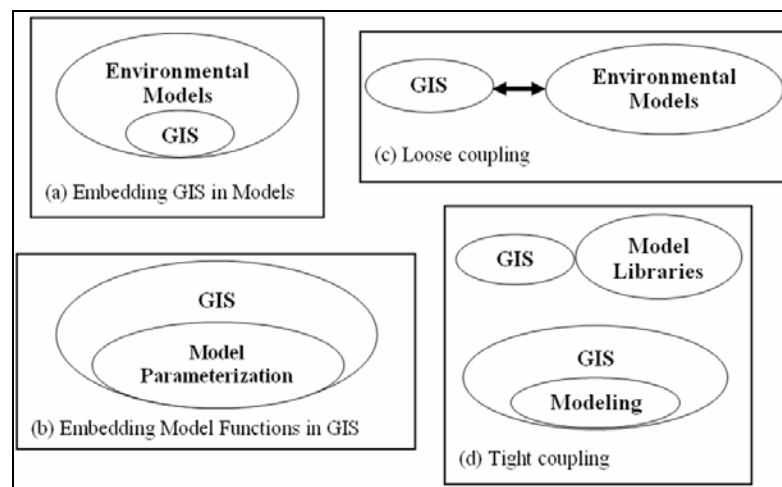


Figure 2.1 Integration of GIS and Environmental Models

Developers of almost every environmental model have realized not only the relevance and need of high-level spatial visualization of their outputs but also the need to generate output formats readable by commercial GIS software. As a consequence, many models have introduced “Export to GIS” capabilities to improve visualization and enhance data sharing properties. Therefore, under the first approach (Figure 2.1a); GIS functionalities are embedded in preexisting modeling systems. This might be termed the conventional GIS post- processing approach in which the classical role of GIS as a georeferenced mapping tool is

implemented. Under this scheme, GIS is considered as a mapping tool and is conceptually irrelevant to the fundamentals of hydrological modeling (Sui and Maggio, 1999).

Under the second approach (Figure 2.1b) and restricted by the static nature (time invariance) of existing GIS, some type of hydrological modeling functions have been added as add-ins or extensions to the most common GIS software packages. These “modeling” capabilities are usually intended for model configuration and parameterization and normally take advantage of spatial analyst extensions to generate hydrologic related data sets that are used as input for many of the industry standard modeling systems.

Some commercial GIS packages have embedded some type of modeling capability in its systems. The ArcHydro data model and associated toolset for water resources in ArcGIS (Maidment, 2002) and the hydrologic functions in the Raster GIS GRASS (GRASS, 1993) are some examples of this approach.

The loose coupling approach (Figure 2.1c) tries to communicate between a standard GIS package and Hydrologic and Hydraulic (H&H) modeling systems via a data exchange framework. Data exchange is normally done through generation of model-specific ASCII or binary data formats that the next system in the workflow can assimilate. Thus, data are transferred and exchanged between models and GIS, with each system having its own way of looking at the data. Data conversion between models and GIS is normally tedious and error prone. Recognizing this fact, standard data exchange formats are now being developed that will facilitate data exchange in more generic and comprehensive forms using

generic and self-described formats, such as XML (Djokic, 1995). Under this scheme, some kind of communication/integration between standard systems is envisioned but ways to centrally handle this integration were not available until recently with new GIS and IT developments.

Under the loose integration approach each external application (including GIS) remains as an independent system that is executed through its own interface after the previous application has produced its output and some data exchange process is executed to enable the next application to use it as its input. In other words, there is no central system accessing the components and the integration depends only on flows of reformatted data from one application to the other.

Many GIS pre- and post-processors have been developed to interface with the industry standard environmental models; some examples are the Watershed Modeling System (Nelson, 2000), the Watershed Analyst, HEC-GeoHMS (USACE, 2000b), HEC-GeoRAS (Djokic et al., 1992, Ackerman et al., 1999, USACE, 2002a), CRWR-PrePro (Olivera and Maidment, 2000) etc., which generate the needed files for specific model configuration and set up. Attempts to synthesize several available GIS-based tools for digital floodplain analysis have also been undertaken (Anderson, 2000).

The Tight coupling approach (Figure 2.1d) promotes the “GIS modeling dream” in which the customization environment of a standard GIS software package is used to develop a complete hydrologic, hydraulic, or environmental system fully inside the GIS hosting environment. By this approach, a user can develop his/her own modeling libraries inside the GIS system. To support this

approach, not only a well-defined interface to the GIS data structures is needed but also a time varying character of the GIS system that meets inherent storage demands is required. Under this approach the environmental models are executed directly from inside the GIS system either by calling the modeling libraries on demand or by directly having model constructs in the GIS structure.

Under the partial tight integration each application is still independent of each other but centrally managed from a single application that can perform calls to execute the external applications as required by the workflow. Many believe this approach is best due to the incorporation of proven and benchmarked proprietary modeling systems that are already widely used and accepted by the engineering community.

Partially tight integrations of ArcGIS with hydrologic libraries like HEC LibHydro have been developed to capitalize on pre-processing and visualization capabilities of GIS (Whiteaker, 2003). A recent fully tight integration for the TOPMODEL hydrological model and GIS (AVTOP) was reported (Huang and Jiang, 2002) in which an already existing stand-alone model was re-developed within the GIS environment by means of the macro language Avenue of ArcView 3.x.

The previous survey of GIS-based integrating approaches exposes several ways of exploiting the GIS benefits in modeling exercises. In order to streamline and automate the generation of flood inundation maps a tight coupling approach (Figure 2.1d) is needed in which stand-alone engineering models are either

externally executed from a central framework or fully re-developed inside the GIS framework.

A fully tight integration (Figure 2.1d Lower) in which all the needed stand-alone models are re-developed within the GIS environment is currently possible only for simplified conceptual models. Conceptual models do not have a physically-based definition that relies in solutions of partial differential equations which are still difficult to formulate and solve inside a GIS environment.

A partially tight integration (Figure 2.1d Upper) that utilizes stand-alone applications and centrally managed from a GIS-based application is selected in this research as the most suitable method to accomplish the proposed modeling integration. This selected approach allows the incorporation of proven and benchmarked proprietary modeling systems that are already widely used and accepted by the engineering community and regulating agencies in flood studies.

The selected partially tight integration provides a framework to centrally handle the modeling integration via new GIS and IT developments. The central system externally accesses the interrelated components and the integration depends on flows of reformatted data that is automatically relayed from one application to the other. That is, the Map2Map application developed in this research fits into the partially tight coupling category (Figure 2.1d Upper).

2.2 GIS-BASED FLOODPLAIN MANAGEMENT

Given that river and floodplain aspects of floodplain management have a spatial component, a GIS-based approach is suitable to manipulate and visualize the spatial distribution of flood project components. Traditionally, GIS overlay

functionalities and computational engines have been used in automated floodplain delineation systems. Several automated GIS-based floodplain delineation systems have been developed to support flood damage assessment components (Noman, et al., 2001). Some of the most well-known systems are: Arc/Info MIKE11-GIS, Arc/Info Floodplain delineation, ArcView MIKE11-GIS, Watershed Modeling System, flood mapping functionalities in FLOODWAVE, and the HEC-GeoRAS post-processing delineation.

In contrast with the abundant systems for automated floodplain delineation, GIS-based flood damage assessment systems have not proliferated. By the end of the 1980s, many countries started to experience a worsening trend in recurrent flood problems mostly attributed to urban and land use developments that substantially change runoff characteristics and drainage configurations. Some of the attempts to address the increasing pattern of severe flood occurrences by means of GIS-integrated systems are summarized below to provide a conceptual framework for the present research.

To formulate appropriate floodplain management strategies in the form of basin management plans, the government of Hong Kong started to develop in 1990 a system for flood risk assessment (Brimicombe and Bartlett, 1996). The proposed flood risk assessment system was based on the transfer of GIS-based parameterization to stand alone hydrologic/hydraulic modeling systems whose output is passed back to GIS for output visualization and reporting. The spatial extent of flood was superimposed to land used configurations to define flood hazard maps as the main foundation for a spatial decision support system. Even

though GIS-based, the system does not represent a true integration of GIS and modeling with central execution of all the involved processes. This system achieves integration by means of data exchange only and not by means of a central and unified execution of chained systems representing the modeling workflow of processes and data.

An early attempt to integrate “industry standard” hydraulic numerical modeling and geographic information systems was done through the ArcView GIS software (Muller and Rungoe, 1995). An interface between the 1-D numerical hydraulic model of the Danish Hydraulic Institute, MIKE 11 and Arc View 3.x was developed, the MIKE11-GIS ArcView interface. MIKE 11 was coupled with ArcView to generate 2D and 3D water level and flood inundation maps. The system allows for rapid generation of inundation boundaries for different flood scenarios, including scenarios with or without flood protection measures. It provided a systematic protocol for locating the inundated land under alternative mitigation strategies. The system allows for making multiple runs and testing a number of scenarios efficiently.

Almost simultaneously with the previous approach in 1996, the Delft Hydraulics research institute developed a flood hazard assessment model for the river Meuse case study in south Netherlands as a direct response to the flooding events of December 1993 (Jonge et al., 1996). Recognizing the fact that the important river related aspects of flooding and managing floods (safety, agriculture, industry, etc.) all have a spatial component, the GIS package

ARC/INFO was selected at the time as the central framework to develop the model (Tineke De et al., 1996).

GIS-based flood hazard assessment schemes have been proposed by several researchers. A flood impact assessment system based on land use was developed using ARC/INFO and its customization language AML (Arc Macro Language) (Boyle et al., 1998). The GIS interface included hydraulic simulations, generic damage curves, and simulation functions for alternative plan evaluations. This method does not have explicit consideration of uncertainties and the floodplain modeling phase starts at the hydraulic level.

A GIS-based flood information system for the Chia-I County in Taiwan has also been reported (Yang and Tsai, 2000). This system encompasses three main components: floodplain modeling, flood damage, and flood information support. The floodplain modeling method uses modeling cells defined by irregular polygons associated with the main channel and to the floodplains. The cell boundaries are defined by high ground corresponding to stream banks, levees, roads, etc. and the land use within each modeling cell is considered to be homogeneous. The downside of this system is the lack of hydrologic simulations based on the design storms defined for flood damage calculations.

More recently, a flood warning and response system that employs aerial photography, terrain elevation data, channel geometry, demographic and structural data, and transportation systems with a hydraulic HEC-RAS model has been implemented to create an automated flood mapping application using the

Geographic Information Systems, ArcGIS, for the Susquehanna River in Pennsylvania (Ackerman, 2004).

Out of the above reported GIS-based systems, only the Hong Kong (Brimicombe and Bartlett, 1996) and the Taiwanese (Yang and Tsai, 2000) approaches included some type of hydrologic and hydraulic integration to support floodplain delineation and flood damage evaluations. The MIKE11-GIS Arc View interface configures an integration scheme with the 1D hydraulic model of the Danish Hydraulic Institute, the Susquehanna flood warning system represents integration with just the HEC-RAS hydraulic component.

All the reported GIS-based Flood Information Systems described above, except for the Susquehanna system, use some level of aggregation for the floodplain infrastructure inventory. The system for Taiwan defines modeling cells with homogeneous land use in it configuring a damage aggregation at the cell level. The case study for Ontario directly divides the floodplain into four types of land uses (residential, commercial, industrial, and open space) without aggregating them at the modeling cell level. Once a property is classified in one of this four land use categories they are all treated the same way in terms of flood damage calculations. Thus, there are two levels of aggregation, the land use level (for flood damage calculation purposes) and the modeling cell level (to simplify hydraulic calculations) which includes the first one also. By having the second level of aggregation for hydraulic reasons the system does not account for land use variations within each modeling cell which represents an important limitation for the flood damage assessment component and its very spatial nature.

The aggregation of floodplain inventories into coarse land uses allows for the use of case specific flood depth-damage curves obtained for each land use through field studies and floodplain inventories and representing one of the three basic functions (as described in section 2.1) needed to perform the traditional economic evaluation for flood damage assessments. These empirical depth-damage curves are normally developed by a property survey of the floodplain and by individual or aggregated estimates of depth versus damage for each land use category in the floodplain.

The HEC-FDA system provides for a somewhat less aggregated approach to evaluate the depth-damage relationship. In it, each structure is given a particular depth-damage assessment based on a more detail definition of the properties' first floor and ground levels. Specification of first floor stages and beginning damage depth stages for each property in the inventory allow for a more realistic approach. However, the depth-damage relationship is aggregated at each damage index location station and the effect of a given depth at the index location relies on very good quality field surveys and evaluations. This approach even though more realistic relies on very hard to get depth-damage relationships that can quickly become obsolete given the dynamic nature of floodplains regularly affected by new regulations, alleviation plans, and changing land use configurations. So, even though the HEC-FDA provides a distributed approach for definition of depth-damage curves (based on a distributed structure inventory) it aggregates the spatial inventory of the floodplain into reach index locations configuring a lumped damage assessment methodology.

By defining the three basic functions at damage reach index locations a supposedly uniform floodplain section gets aggregated to each index location station. Through this approach any changes on the floodplain configuration are difficult to introduce and will imply a new definition of the basic functions.

As part of this research, an alternative approach for flood damage assessment is sketched that takes full advantage of the distributed strength of GIS to obtain the depth-damage relationship at each structure based on the current depth at that spatial location as given by the integrated floodplain delineation process (The Map2Map application). By doing this, a more realistic assessment is expected that may keep up with the dynamic development of the floodplains without having to redefine the aggregated depth-damage curves at the land use, modeling cell, or index location level.

2.3 FLOOD MITIGATION PLAN EVALUATION AND SIZING

After the floodplain has been simulated and the flood inundation extent delineated under a given floodplain and hydrologic configuration, some framework is normally used to evaluate a proposed set of flood management plans. To accomplish a flood plan evaluation flood maps are normally generated based on GIS overlay analysis. In GIS-based systems the flood maps are then used for flood damage calculation processes based on three basic functional relationships for hydrology (flood-frequency curve), hydraulics (rating curve), and economics (stage-damage curve).

The evaluation process of a flood alleviation project follows a standard simulation algorithm based on standard and approved hydrologic and hydraulic

software. Normally, the simulation of local projects composed of levees or similar structures allows one to determine if the proposed structure protects against a given design flow event as required by federal or local regulations. In particular, the levee certification process was traditionally driven by protection criteria against river stage associated with the 1% chance flood (100 yr flood event) plus 3 feet of freeboard. This conventional approach based on flood frequency analysis was abandoned in the early 1990's when the Corps of Engineers adopted risk analysis techniques that replace the aforementioned "1% event plus freeboard standard". Besides having to consider the set of components that make up an optimal solution some authors have tried to answer the question of optimal size of already selected projects.

Multiple criteria can be used to define the "optimal" flood control solution. One alternative is represented by the economic efficiency of the project. The optimal sizing of flood damage reduction plans based on economic efficiency has been addressed by a combination of a simulation and an optimization models. Methods for optimal sizing of flood damage reduction plans based on economic efficiency have been reported (Wurbs, 1996). The method searches for an optimum mitigation plan that may include non-structural measures. In these optimization approaches, the objective function is normally represented by the minimization of the total system cost and decision variables are represented by the size of the structural components of the system.

To develop a flood impact analysis system, flood related information describing the spatial distribution of the floodplain is needed. GIS layers

containing an inventory of damage sensitive facilities that belong to some category of land used (residential, educational, industrial, commercial, etc.) are obtained. Each of these damage sensitive components must have an associated database that provides the attributes on which damage estimates can be calculated using economic methods such as benefit-cost analysis.

2.4 PROPOSED CONTRIBUTIONS TO KNOWLEDGE

The idea of model integration has been described under various statements as “bringing the models together”, “watershed in a box”, or “Regional Watershed Modeling System”. As mentioned before, attempts to accomplish this modeling integration have been reported in the literature which follows a step-wise approach in which the data is passed from one modeling program to another in a manual fashion favoring a time consuming and error-prone workflow that is functional but not very productive nor efficient. The GIS-based character of the reported attempts is represented only by the transfer of GIS-computed parameterizations to stand-alone modeling systems whose output is passed back to GIS for visualization purposes only. Thus, this step-wise integration based only on pre- and post-processing data exchange does not represent a true integration of the involved modeling systems.

Process analysis/models represented by stand-alone simulation models may synergistically employ the output of another simulation model and thus be sequentially linked. To achieve the linkage, geographically oriented data models can be used to spatially associate the exchange of information from one process model to another. Once all the needed integrating elements are developed, a

workflow model (herein in a GIS environment called ModelBuilder) can be used to empower a seamless end-to-end integration as the gluing platform that prevents the conventional step-wise approach.

The true geographically-driven integration proposed in this research empowers a spatial linkage of all the necessary flow exchange points (hydrograph entry points) via a connection based on ModelCodes. For HEC-HMS the HMSCode is represented by the name of the hydrologic elements entering a runoff hydrograph to the stream network and geographically represented in GIS. For HEC-RAS a RASCode constructed with the union of the StreamID, ReachID, and Station is proposed to establish the connection between GIS and HEC-RAS. This connection makes possible the transfer of information (runoff hydrographs) from HEC-HMS elements to HEC-RAS cross sections via a geographic framework represented by the Arc Hydro data model for water resources.

None of the reported systems (out of only two that include the hydrologic end) allow for readily incorporating rainfall inputs of various natures and forms like historic, real-time, forecasted, and design storm events. The modular character of the proposed integration allows code reuse of common elements and straight forward incorporation of new components to account for several rain map formats potentially used to drive the system. By having this flexibility, it is possible to drive the integrated system with diverse rain maps to generate multipurpose flood maps.

The herein proposed integration system relies on pre-existing modeling systems that might be fully generated for a given study area by means of Interface

Data Models in charge of storing and generating a complete GIS-based model configuration above the traditional and just-geometric pre-processing setups.

Chapter 3 Methodology

3.1 CONCEPTUAL DEFINITIONS

The development of the seamless and integrated system revolves around three main models: Process models, Data Models, and a Workflow Model. The process models represent the physical processes that need to be simulated (rainfall-runoff transformations and streamflow routing in this case), the Data Models support the geographic linkage between the two process models, and the Workflow Model automates the series of tasks needed to convey data among the various components. In addition to these three main models, components dealing with spatially referenced time series exchange need to be included for the integration as well as components for accomplishing the final floodplain delineation.

In summary, the proposed integrated system must be able to acquire NEXRAD map time series to drive hydrologic simulations in HEC-HMS whose streamflow outcome will feed the hydraulic model HEC-RAS, which provides the needed water surface elevations to perform the final floodplain delineation to get the flood inundation map.

Figure 3.1 illustrates the geographic conceptual integration as it applies to Rosillo Creek in the San Antonio River Basin starting with the ingestion of radar precipitation data, execution of hydrologic simulations, transfer of flows through a geographic framework, executing the hydraulic simulations, and delineating the floodplain.

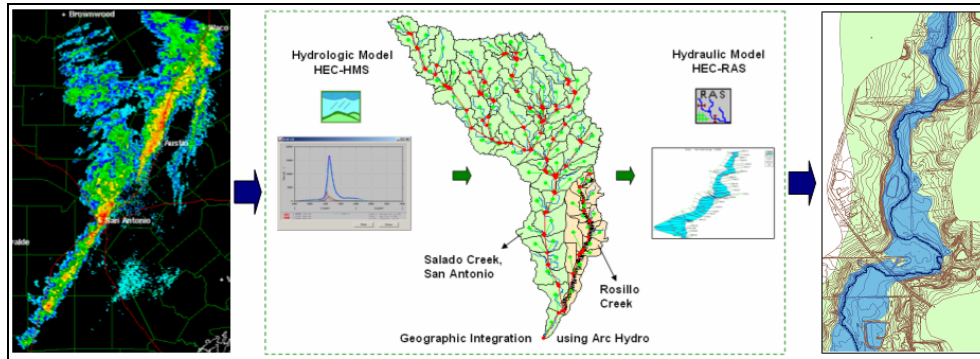


Figure 3.1 Floodplain Modeling Schematic

Figure 3.2 provides the schematic for the above conceptual integration with all the sequential tasks needed to transform a rainfall map time series into a flood inundation map.

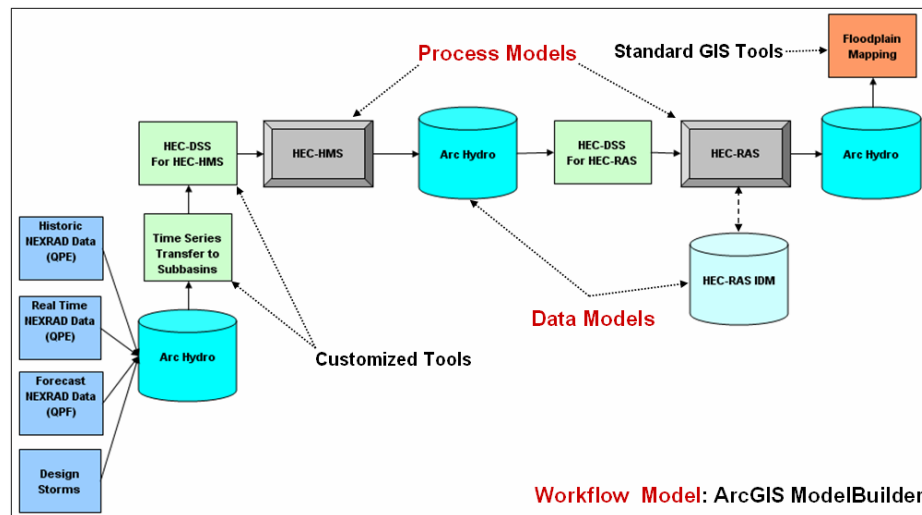


Figure 3.2 Conceptual Elements to Convert a Rainfall Map to a Floodplain Map

Figure 3.2 shows the involvement of the aforementioned three main components: process models, data models, and the workflow model. Each of the

main conceptual components for the integration system is described in this chapter as well as the case study used for the pilot implementation and proof of concept.

3.1.1 Identifying Modeling Elements

The modeling features in a hydrologic or hydraulic simulation model must be uniquely identified so that input and output time series associated with them can be labeled appropriately. For the purpose of this integration this unique identifier is called the "ModelCode".

In HEC-HMS the identification of modeling elements is done by using the HMSCode, which is a text string used within HEC-HMS to identify each model feature (called a Hydrologic Element in HEC-HMS); in HEC-RAS there is no single identifier used within the RAS model, rather each cross-section is identified by its stationing distance from the upstream end of a river, a river is broken into reaches, which generally flow from confluence to confluence in a river network. An effective RASCode is thus constructed by joining the river, reach, and stationing labels in a comma separated text string.

3.1.2 Geographic Representations

There must be an exact one-to-one match between the geographic representation of modeling elements in the hydrologic and hydraulic model and in the GIS. ModelCode attributes must be attached to the corresponding Arc Hydro features.

For HEC-HMS, the geographic match is accomplished by using the Arc Hydro schematic network, and by associating HEC-HMS Hydrologic Elements,

either with Schematic Nodes (for Subbasin, Junction, Source, Sink, Diversion and Reservoir elements) or with Schematic Links (for the Reach element).

For HEC-RAS, the geographic match is accomplished by having the Arc Hydro feature class, CrossSection; correspond exactly to the cross-sections used in the RAS model.

3.1.3 Flow Change Points

Flow change points are where time series information is exchanged between the hydrologic and hydraulic models. These points are represented in Arc Hydro as HydroJunctions.

Relationships must be formally populated between the HydroJunction feature class and the related Arc Hydro feature classes matching the model topologies and on which ModelCode attributes have been populated (Schematic Node for HEC-HMS and CrossSection for HEC-RAS). There is thus a one HMS to one Arc Hydro to one RAS linkage that acts as a path for time series transfer between the georeferenced flow change points.

3.1.4 Arc Hydro Time Series Extension

A geodatabase structure in Arc Hydro that exactly mimics HEC-DSS components must be created so that the ArcGIS geodatabase can contain a mirror image of the time series information for model elements stored in HEC-HMS and HEC-RAS (which share HEC-DSS as their time series data storage system). This means two related tables, a catalog and a time series table.

3.1.5 Time Series Retrieval from HEC-DSS

A software application must interpret the HEC-DSS time series catalog and know which time series records apply to a particular model feature. This is done for HEC-HMS by means of the hydrologic element's name stored in the "B" pathname part in DSS and with ownership on the time series data. For HEC-RAS the DSS time series retrieval is done by means of the entire DSS pathname which is referenced in the flow file by a software application in charge of updating the input files for HEC-RAS.

3.1.6 Automated Workflow

An automated workflow sequence must be created to sequentially execute the tasks needed to accomplish the hydrologic and hydraulic model executions and the data flow between them and the GIS. This is done with a customized implementation of the ArcGIS Model Builder called Map2Map, which includes in part tools that are generic to the GIS and in part customized tools that were created in this research.

3.2 CASE STUDY: THE SAN ANTONIO RIVER BASIN

This research is being developed for the San Antonio River Basin (4,000 sq. miles) in Texas, which comprises a diverse basin including regulated subbasins and complex urban drainage systems that flow south-east towards the Gulf of Mexico.

A subbasin of the San Antonio River basin, Salado Creek (222 sq. miles) and its tributary, the Rosillo Creek (29 sq. miles) were used as the pilot basins due to the availability of 2-foot contour lines from land surveys to characterize the

channel cross sections, available HEC-HMS (USACE, 2000a) and HEC-RAS (USACE, 1997) hydrologic and hydraulic modeling systems respectively (as provided by the city of San Antonio), and for representing complex urban settings in the city of San Antonio.

Since late 2002, the San Antonio River Basin is the subject of an ambitious modeling project to provide engineering support to evaluate different planning and management alternatives for flood control plans and flood scenarios. For the proposed regional watershed modeling system of the San Antonio River Basin, management alternatives can be studied by employing the modeling capabilities herein develop and to assess the change in water surface profiles due to channel improvements and levees. The model will allow visualization of multiple plan analysis performance to aid in the task of ranking projects to be presented for local government approval and execution. In general the project's main objective is to provide tools for better storm water management with emphasis on flood regulation and control.

The San Antonio River Basin contains a diverse basin composed of regulated sub basins and complex urban drainage systems. Significant areas of the basin are expected to undergo land use changes whose effects also need to be evaluated.

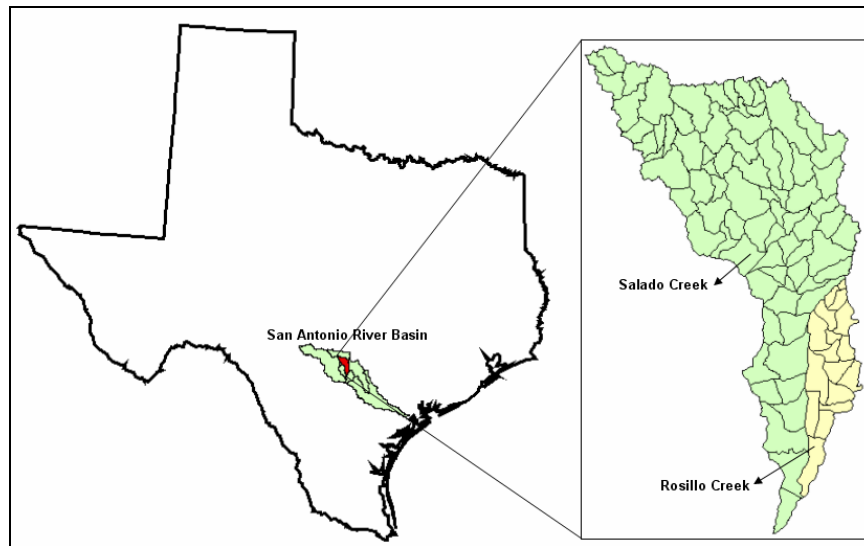


Figure 3.3 Relative Location of Salado and Rosillo Creeks in Texas

3.3 INPUT DATASETS

3.3.1 Precipitation Datasets

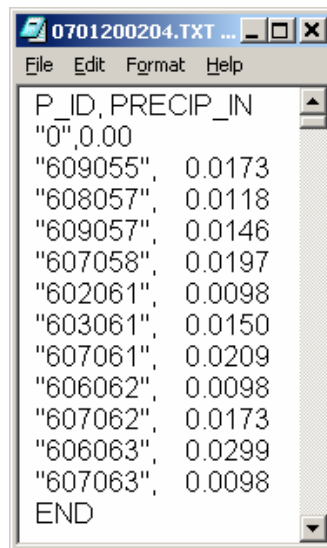
Rainfall hyetographs need to be ingested into the proposed integrated system to drive the hydrologic simulation of the study area. Better radar precipitation estimates at higher temporal and spatial resolutions represented by NEXRAD products are now available through the National Weather Service to provide historical and real-time information for the proposed integration.

Thus, the transfer of time series records between systems having different formats, a selected NEXRAD format and the HEC-DSS data format, is needed. The modeling systems of the Hydrologic Engineering Center (HEC) use the HEC Data Storage System for time series, HEC-DSS (USACE, 1995), a database specifically designed for water resources applications that uses a block of time

sequential data, called pathnames, as the basic unit of storage and that stores the records in binary format files for access efficiency.

A subset of NEXRAD time series data for Texas was provided by the Guadalupe Blanco River Authority in Texas as a series of ASCII files. The subset was comprised of 552 hourly text files for a storm event in late August of 2002 and for the entire state of Texas. For the current format being used in the integration system, the rainfall data are stored in individual ASCII data files named after the hourly time stamp they contain.

A snippet of the file content for one of the text files is shown in Figure 3.4 for filename “0701200204.” The filename contains the time stamp embedded in it with the date format MMDDYYYYHH. Thus, the filename “0701200204” contains a rainfall data for an event occurring on July 01, 2002 at military hour 04. Figure 3.4 depicts a snippet for this data text file. The text file contains a header defining the first column as a pixelid (P_ID) and the second column as the precipitation value in inches (PRECIP_IN). At the end of the data records, the text file ends with an END statement.



P_ID	PRECIP_IN
"0"	0.00
"609055"	0.0173
"608057"	0.0118
"609057"	0.0146
"607058"	0.0197
"602061"	0.0098
"603061"	0.0150
"607061"	0.0209
"606062"	0.0098
"607062"	0.0173
"606063"	0.0299
"607063"	0.0098
END	

Figure 3.4 Radar data in ASCII format

The precipitation values on the above Figure are associated with a PixelID pointing to a NEXRAD cell (4 km by 4 km square grid cell) in the mappable HRAP format of NEXRAD. The layer containing the HydroResponse feature class in Arc Hydro having the NEXRAD cell definition was provided by the Brazos River Authority in Texas. This layer is a statewide NEXRAD HRAP grid in which the “Rfcthsn_ID” field in the grid layer corresponds to the P_ID field (first column Pixel ID) in the above data text file. Figure 3.5 shows the 873 HRAP cells covering the San Antonio River Basin (about 4,000 sq. miles).

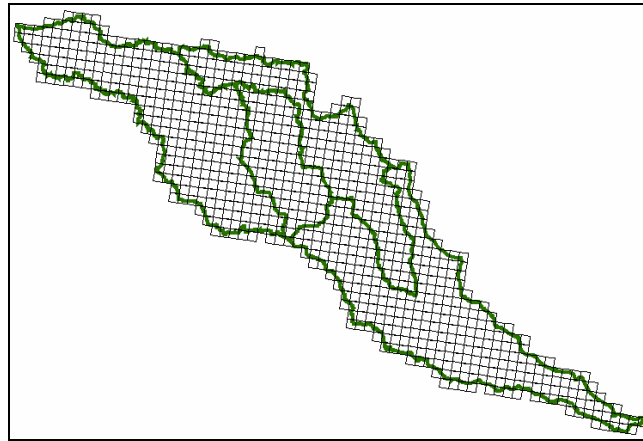


Figure 3.5 NEXRAD square cells (873 cells) covering the San Antonio River Basin, Texas

NEXRAD radar precipitation data for the Hydrologic Rainfall Analysis Project (HRAP) grid (a 4x4 Km square-celled map grid) is read from a series of text files containing hourly data for a any given storm event. The text files are read and the time series are transferred to geodatabase tables with Arc Hydro format (TimeSeries and TSType tables) as well as a HEC-DSS catalog to store the HEC-DSS descriptors of the data and to enable transfer of this time series data into the HEC-DSS system.

3.3.2 Hydrologic and Hydraulic Models

The execution of Map2Map for the processes involved in generating a flood inundation map from a rainfall map relies on having existing, calibrated, and validated hydrologic and hydraulic models. For the current development existing versions of HEC-HMS and HEC-RAS are used to serve as the main transformation processes of the workflow. For future developments, the possibility to include other widely used models could be added, like MIKE 11 or

MIKE 21 (DHI, 2000; 2001) hydraulic models used by the US Bureau of Reclamation for management operations on areas west of the Mississippi river or the FLDWAVE hydraulic model (Fread and Lewis, 1988) used by the National Weather Service in forecasting operations. Further developments to include combined one and two dimensional routing models (Runge et al., 2003) and even three dimensional models (Sinha et al., 1998) might be considered as a follow-on research component.

3.3.3 Geographic Representations in Arc Hydro

The relevant stream and floodplain related aspects of flooding and its management have a spatial component. Therefore, a GIS-based system is a valuable component in developing an information system (if not as a central platform, at least as a component of it). This component contains a base geographic representation and description of the water features of the landscape. The spatial representation of the basin's components is further enhanced if created under a geodatabase data structure formatted with a predefined data model for water resources. The Arc Hydro Data Model for Water Resources (Maidment, 2002) represents a based dataset enhanced with key relationships, attributes, and water resources related parameters upon which a modeling exercise can build upon.

An Arc Hydro data model (or ArcGIS Hydro data model) has been created for the pilot Salado Creek basin to serve as a geographic integrator for the modeling system under development. Thus, the “container” for the streamflow time series values needed for model connectivity with HEC-RAS is a GIS

geodatabase formatted with the Arc Hydro data model for water resources. The selected form of the schema version that defines the data structure in Arc Hydro permits time series storage by means of object classes (tables) with spatial relationships to features that own the time series datasets.

For the integration to be complete it is necessary to capture not only the flow values but also the spatial associations between independent hydrologic and hydraulic elements which have a spatial character thus, the GIS geodatabase system becomes a means of storing the model connectivity elements and their relationship.

Geographic representations of basin hydrology are accommodated by using standard components of the Arc Hydro data model, which are extended to incorporate model-specific elements. Spatial consideration is also needed for key components to exchange information between two external models (HEC-HMS and HEC-RAS) and to accommodate time series datasets and corresponding model-specific descriptors of time series data to enable time series data exchange.

Thus, Arc Hydro data models were developed for the Salado and Rosillo Creeks following the current version of its content in terms of theme groups of feature classes and nomenclature (Maidment, 2002). In addition to the standard content in Arc Hydro the following elements were added to the database scheme:

The SchematicNetwork was extended by including an HMSCode field which was populated with the name of the corresponding hydrologic elements in HEC-HMS corresponding to streamflow entry points of the network.

An additional relationship class was built between the HydroJunction feature class and the SchematicNetwork to allow Arc Hydro to connect with the hydrologic elements represented by SchematicNodes.

An additional relationship class was built between the HydroJunction feature class and the CrossSection feature class to allow Arc Hydro to connect with the hydraulic definition of the river channel geometry at exchange points

The SchematicNetwork in ArcHydro resembling the hydrologic topology in HEC-HMS is shown in Figure 3.6.

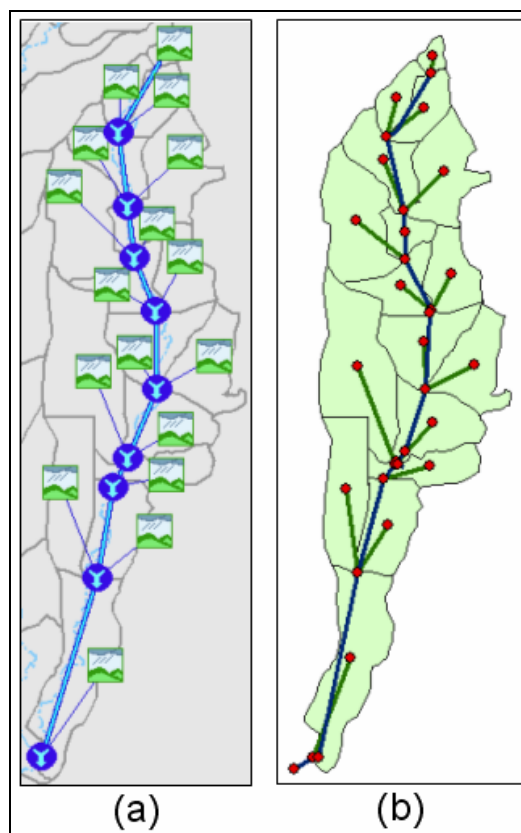


Figure 3.6 Geographic Representation (b) of Hydrologic Model (a)

The CrossSection feature class in Arc Hydro resembling the hydraulic topology in HEC-RAS is shown in Figure 3.7.

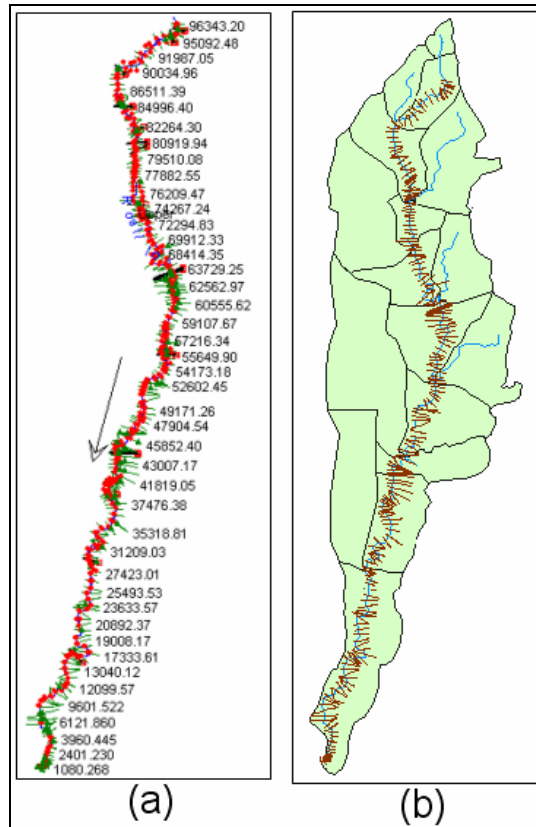


Figure 3.7 Geographic Representation (b) of Hydraulic Model (a)

3.3.4 Terrain Surface Dataset

Flood studies require a very detailed representation of the terrain model (10 m resolution cell size or less) of the river's main channel and lateral overbank sections as well as the inclusion of relevant soft and hard break lines in addition to mass points for the representation of infrastructure (like building and road footprints) over the floodplain.

Public domain digital terrain models normally have low resolution and are not suitable for describing river channel geometry and floodplain boundaries (Werner, 2001) unless more detailed land survey studies are merged into them (Tate et al., 2002). Contours obtained from topographic surveys are a common source of detailed digital elevation data for local and regional studies. For the Rosillo creek, a digital elevation model (ground surface model) was created from field surveys that generated 2-foot contour line maps (developed by contractors of the city of San Antonio) in the form of CAD files (.dwg, dgn, dxf) and covering the extent of the Rosillo creek's floodplain. Conventional GIS Spatial Analyst functionalities were used in this research to convert the contour line map into a DEM (raster or grid-cell representation of the surveyed terrain elevations). The topographic information from this land survey was also used to supply the cross section geometric information needed for the configuration of the HEC-RAS hydraulic model to ensure consistency in the analysis and in particular for the flood delineation's final phase.

3.4 FLOODPLAIN MODELING

Floodplain modeling is supported by hydrologic and hydraulic engineering simulations. A module for data acquisition, model execution, and models connectivity is a central component of the proposed regional watershed modeling system.

When building Flood Information Systems, a great deal of effort goes into hydrologic and hydraulic model configuration. The system proposed here uses the

geodatabase (under Arc Hydro format) as the central data container for each of the included hydrologic model representations.

For this research, the integration of the HEC-HMS hydrologic system (single event rainfall-runoff transformation) and the HEC-RAS hydraulic system (one-dimensional steady flow routing model) of the U.S. Army Corps of Engineers are accomplished. In this way, this development is in tune with one of the two FEMA objectives for the map modernization plan, the Automated Hydrologic and Hydraulic Engineering objective for taking advantage of new technology to automate mapping products of the National Flood Insurance Program (NFIP). In particular, the selected models for this integration (HMS and RAS) comply with the National Flood Insurance Program criteria and thus are accepted for use in the flood hazard mapping program of the NFIP (Noman et al., 2001).

Both models, HEC-HMS and HEC-RAS, with appropriate configurations for the Salado and Rosillo Creek respectively were provided by the engineering division of the City of San Antonio. Therefore, the two models did not have to be generated for the proposed integration as these previously generated and calibrated models were official and available.

3.4.1 Transferring Time Series to HEC-DSS

Once the precipitation records have been reassigned to the watersheds they are transferred from geodatabase time series tables to HEC-DSS binary format based on the HEC-DSS Time Series Type table containing the needed arguments

3.4.4 External Execution of HEC-RAS

A previously set HEC-RAS model for the river system is called with reference to the previous flow change location records. The execution of HEC-RAS is done through the HEC-RAS object library that exposes some execution and writing options.

3.4.5 Automated Floodplain Delineation

Flood mapping also plays an important role in floodplain analysis. The task of generating a flood depth and flood extent map relies heavily on the quality of the previous modeling phase as well as on the quality of the available terrain model of the floodplain (Digital Elevation Model).

The simulated volumes of water must follow flow paths that honor the natural and artificial barriers over the floodplain. For this, adequate modeling and terrain representations of the stream-floodplain system must be available to carry on reliable standard and automated methodologies for floodplain delineation.

3.5 INTEGRATING ELEMENTS

It is a common engineering practice to use a hydrologic model (i.e., HEC-HMS) to compute the runoff from a watershed and then use a hydraulic model (i.e., HEC-RAS) to compute the resulting water surface profiles with great level accuracy for use in flood related studies, like FEMA's flood insurance studies (FIS) (Bedient and Huber, 1992). Under the Map2Map application framework, HEC-HMS and HEC-RAS simulations are integrated through flow exchange points georeferenced in a GIS framework, and executed in an integrated sequence using the new visual programming environment for batch geoprocesses, the

ArcGIS ModelBuilder. Thus, the integration of the two external models is done through Arc Hydro and implemented under a GIS programming environment meant for connecting only GIS geoprocesses but here extended as a hybrid connection of standard GIS geoprocesses and user defined script tools for executing non GIS related tasks.

By putting all of these integrating components together, the developed system for the linkage process can take an arbitrary HMS or HEC-1 model simulation and use it to drive the computation of water surface elevations under a geographically connected framework. The conceptual schematic of this conceptual integration is shown in Figure 3.9.

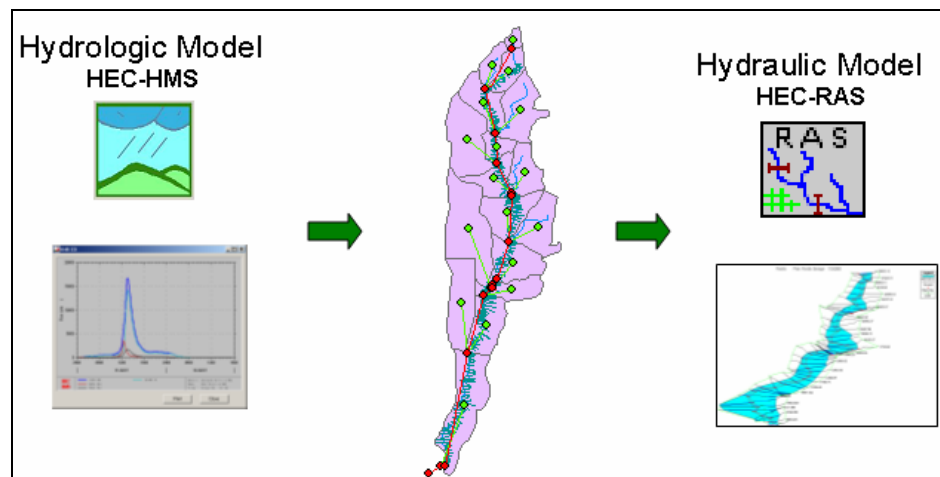


Figure 3.9 Simulation Models Integrated Through Arc Hydro

3.5.1 The Hosting Geographic Environment

Different levels of developments have been accomplished so far that use Geographic Information Systems (GIS) as a tool for water resources research and

analysis. In particular, GIS has been used to create databases of spatial information, to explore spatial relationships between related elements, and to exploit those spatial relationships to perform three main tasks: spatial parameterizations, data models for water resources, and integrated modeling environments for water resources.

The developed GIS-embedded integration for hydrology and hydraulics is herein called the Map2Map application and was built as a modular system (not a single monolithic program) that utilizes separately compiled subprograms (in the form of DLLs) referenced by Script files in charge of simple tasks (like passing arguments and performing calls to the DLLs) that can be loaded as components in the new visual programming environment for batch processing of ArcGIS 9, the Model Builder. By using the ModelBuilder as the gluing workflow model, stand-alone engineering models can now be incorporated as tools inside the ModelBuilder framework and combined with other customized and standard tools to accomplish complex tasks.

In broad sense, the application consists of a main hosting platform, the Model Builder (visual environment for building spatial models in GIS), and many independently compiled subprograms as well as standard GIS tools, linked together to form one single executable run-unit for the transformation of rainfall maps to floodplain maps.

ModelBuilder can be defined as a visual environment for assembling workflow models of geoprocessing tasks. It allows the creation of geoprocessing models made up of organized, compatible and connected individual tasks

(standard or user-defined) creating workflow sequences that perform more complex tasks.

As mentioned before, a GIS-based Flood Information Systems is deemed suitable as the central controller provided that it has adopted the pieces of technology to enable it, in particular Object Orientation and Relational Database Management System. With current technology, the GIS Integration Development can be executed into alternative ways, through toolbars and command buttons or through the newly developed “Model Builder” functionalities of ArcGIS 9.0.

With each option having its own pros and cons, the coupling with engineering models via these approaches facilitates a modular development that promotes reuse and straight forward maintenance and updates on each module of the interface system. The main model linkage components to be included in this GIS system are listed below.

3.5.2 Mapping Time Series to Watersheds

The Time Series records associated with NEXRAD cells are map to the watersheds to generate a hyetograph to each hydro response unit in the rainfall-runoff transformations in HEC-HMS. The time series transfer operation is done by means of a previously created intersection layer generated from the NEXRAD and the watershed polygons. By computing the NEXRAD cell area inside a given watershed polygon the tool estimates the average rainfall over the watershed (aerially weighted hyetographs).

3.5.3 Model-Specific Elements

A geographic description of the features represented in the hydrologic and hydraulic models is needed. In the case of HEC-HMS, this means a geographic representation of the seven feature types that can be represented in a basin file (sub basin, reach, source, sink, diversion, reservoir, and junction).

Each of the model-defined features has a name that identifies the element inside the model's structure. In order to achieve the proposed geographic integration the feature and its name as defined in the existing model needs to be known to the spatial representation of the model's topology (the Arc Hydro data model). Thus, besides having the spatial features in GIS a new attribute field containing the features name (ModelCode) needs to be created and populated based on the model's notation. For HEC-HMS the ModelCode is called the HMSCode and for HEC-RAS it is called the RASCode.

Arc Hydro provides functionality to construct a Schematic Network that resembles the HEC-HMS conceptual topology of hydrologic elements. This schematic network creates a geographic match between the GIS geodatabase and the HEC-HMS model features. Additionally, the schematic node, a component of the schematic network, is used to denote the outlet nodes in HEC-HMS by populating them with the specific HMSCode, to enable the connectivity of the models. Figure 3.10 illustrates the schematic network for the Salado and Rosillo Creeks which closely follows the HEC-HMS conceptual configuration is shown.

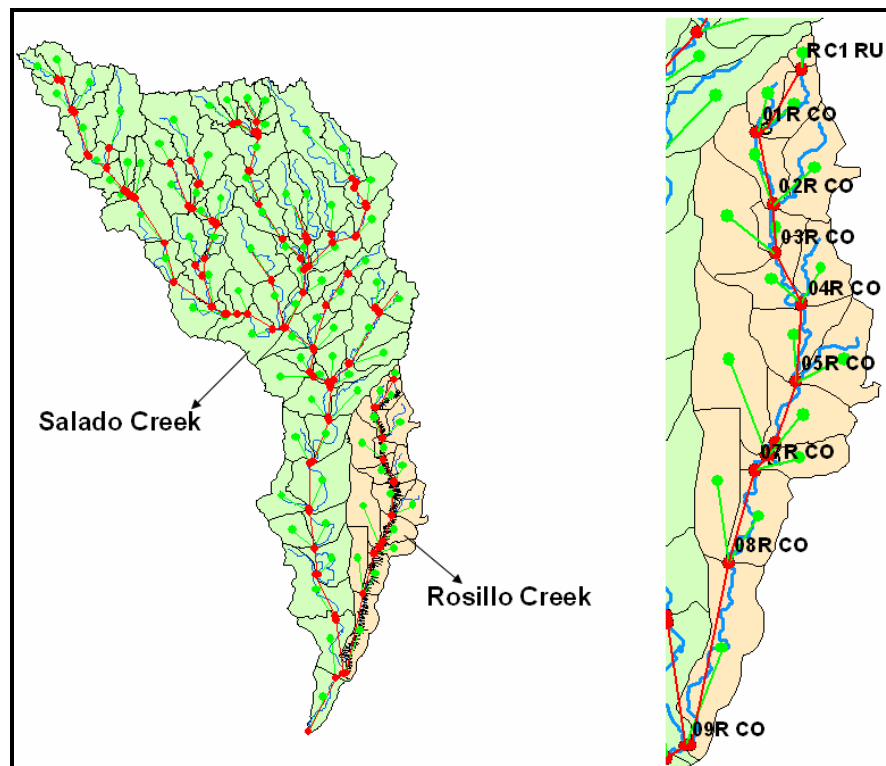


Figure 3.10 SchematicNetwork for Salado and Rosillo Creeks

The Arc Hydro SchematicNetwork (in Figure 3.10) composed of the SchematicLink and SchematicNode feature classes has been used to create a representation of hydrologic elements and its connectivity as it is defined inside the HMS model itself. The equivalent topologies in Arc Hydro and HEC-HMS are shown in Figure 3.11. For Rosillo Creek, HEC-HMS has 35 hydrologic elements (17 subbasins, 9 reaches, and 9 junctions). Out of these 35 hydrologic elements, only 10 model features (1 subbasin and 9 Junctions) are being used as flow exchange points or streamflow entry points (RC1 RU, 01R CO, 02R CO, 03R CO, 04R CO, 05R CO, 06R CO, 07R CO, 08R CO, and 09R CO).

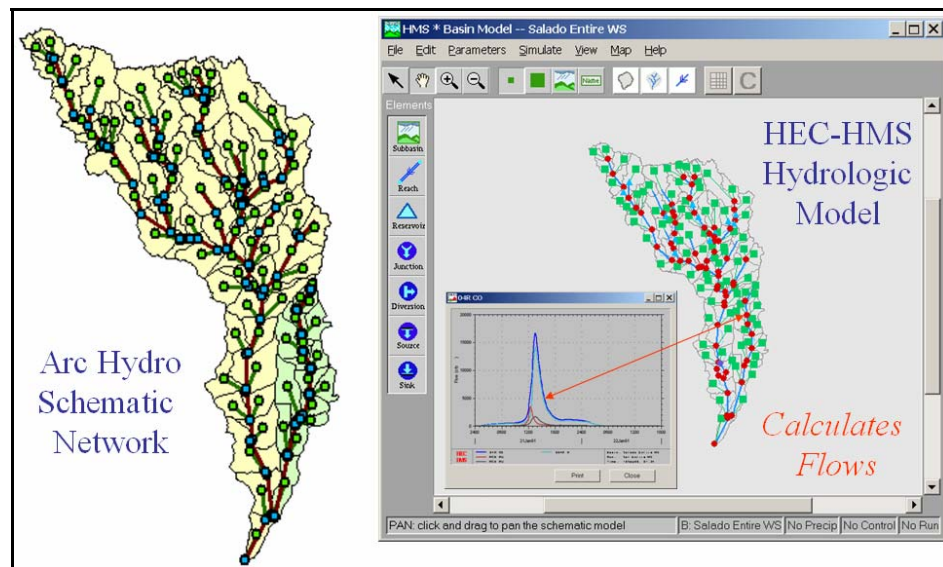


Figure 3.11 Geographic and Hydrologic Correspondence

In the case of HEC-RAS, the spatial cross section elements have a geographic representation by the cross sections of the channel which is also a standard Arc Hydro feature class. All the model-specific feature classes needed for integration are already present in the Arc Hydro data model and can be used to support the spatial linkage between HEC-HMS and HEC-RAS. Figure 3.12 shows the equivalent representation between Arc Hydro and HEC-RAS.

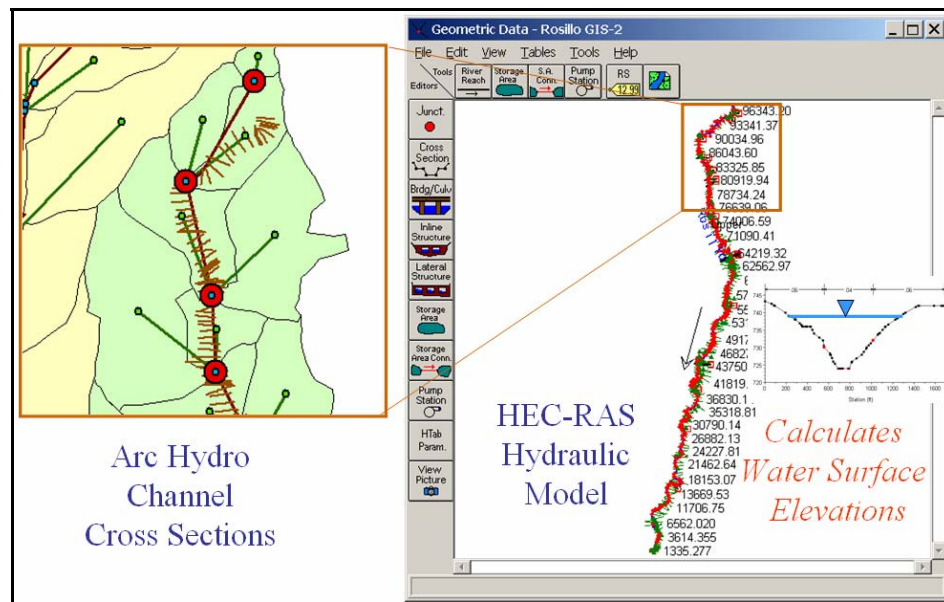


Figure 3.12 Geographic and Hydraulic Correspondence

3.5.4 Model Codification in Arc Hydro

A unique identifier or code representing connectivity features in the models to be linked is needed. For HEC-HMS this code corresponds to the text string used as a feature identifier in the basin file for the upstream hydrologic element that enters streamflow into the river system. These elements normally correspond to outlet-nodes representing tributary entrance points or the headwater watershed for the most upstream river segment over which runoff hydrographs are computed. A list of these elements directly contributing flow to reach segments is also defined in the basin file (.basin file extension) that describes the geometry of the HEC-HMS model. These ModelCodes in HEC-HMS (HMSCode) are stored in the "B Part" of the block pathnames in the HEC-DSS time series file for HEC-RAS.

For HEC-RAS, the connectivity is accomplished by knowing which cross section is associated with the previous hydrologic features (HMSCoded elements). This implies knowing which river, reach and stationing the cross-section lies on, e.g. “Rosillo, Upper, 93528.56”. This identifier represents a sequence of comma-delimited elements herein defined as RASCodes.

Ideally, this model codification should obey a systematic naming convention for watersheds, streams, reaches, etc. to be applied throughout the entire San Antonio basin. A proposed generic naming convention has been adopted by the Florida Water District’s (Southwest Florida Water Management District, 2002).

To enable connectivity between the proposed H&H models, the geographic connectivity features (SchematicNodes and CrossSections) have to be labeled with the corresponding HMSCode and RASCode values. For the HEC-RAS features, the cross sections, this task might be straight forward if the GeoRAS interface was used to create them since the StreamCode, ReachCode, and Measure/Station values are created by the interface and the three values can simply be concatenated to get the three components of the RASCode attribute.

For the HEC-1 or HEC-HMS models that are normally available, this labeling has to be done manually unless some protocol for systematic naming convention of modeling elements is adopted as proposed by the initiatives of the Southwest Florida Water Management District’s procedures manual, in which case the task might be automated.

For this particular case, the exchange of hydrologic information needs to be done at 10 flow change location points representing HEC-HMS outlet-nodes in the hydrologic model and associated with downstream cross sections in the HEC-RAS hydraulic model. Figure 3.13 depicts this case study layout with emphasis on the ten points of information exchange between the hydrologic and hydraulic models.

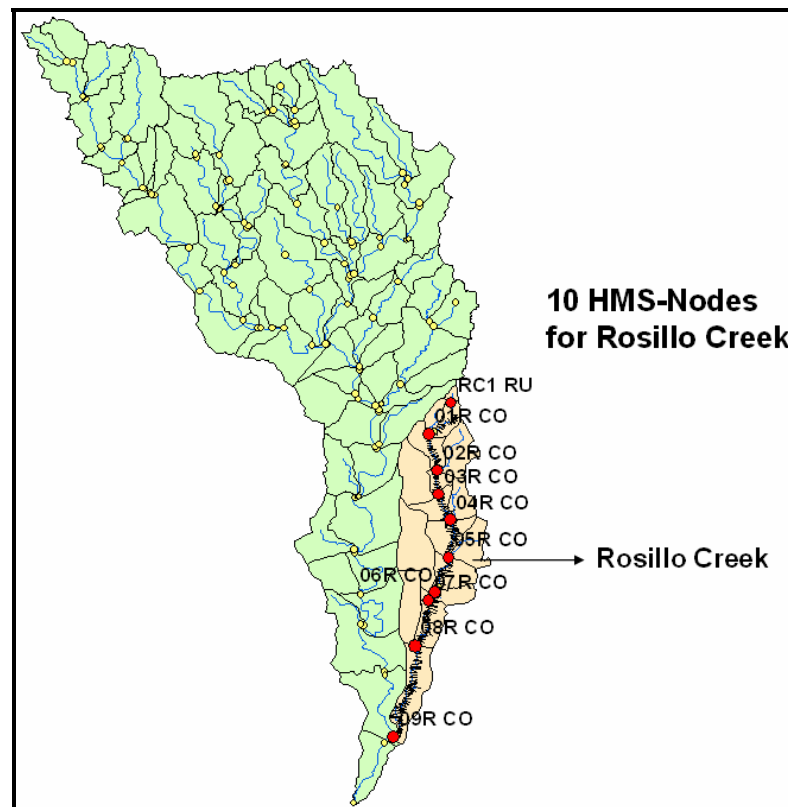


Figure 3.13 Points of Information Exchange in HEC-HMS

These points of information exchange representing hydrologic outlet-nodes in one model (HEC-HMS) are associated with corresponding hydraulic

cross sections in the hydraulic model (HEC-RAS). The two independent but model-related elements (outlets and cross sections) have independent naming conventions (herein called ModelCodes) through which they can be associated. The model independent naming convention is shown Figure 3.14.

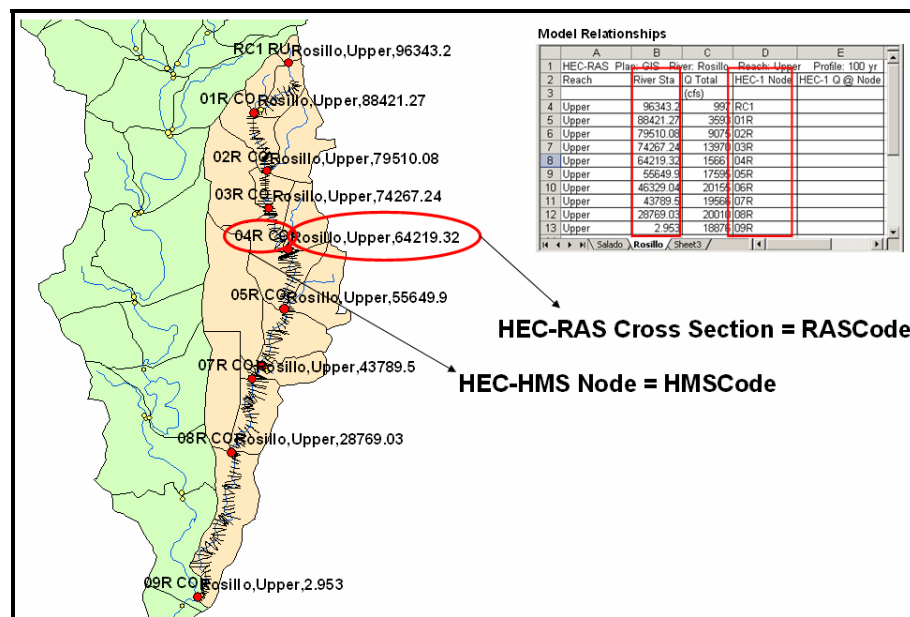


Figure 3.14 Points of Information Exchange in HMS and RAS

3.5.5 Geographic Association of Exchange Points

After the entire Hydrologic and Hydraulic model feature representations have been populated with a model code (HMSCode and RASCode) for potential use as points of information exchange, a relationship class is needed between them to create a 1 to 1 association between the HMS entry runoff nodes and the receiving RAS cross sections. In other words, not all cross sections in the hydraulic model are points of information exchange between models, and each

entry runoff node will have a unique cross section at which transfer of flow information will take place. Figure 3.15 illustrates how the models communicate with one another through Arc Hydro at designated points based on the model codification and relationship classes added to the Arc Hydro framework for the purpose of integration.

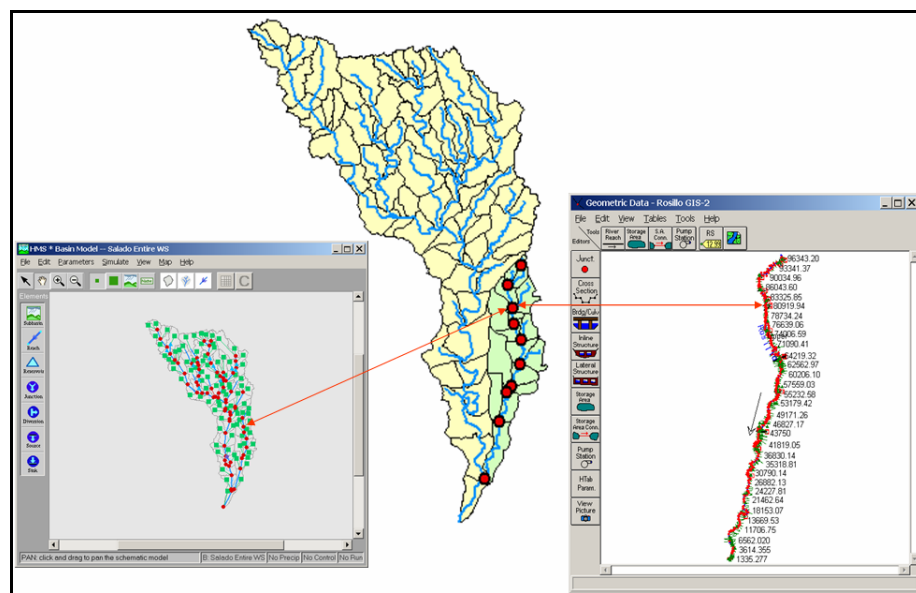


Figure 3.15 Model Connectivity through Flow Change Locations

The definition of "flow change location points" where the models are to exchange hydrologic information represents a task with full spatial character.

In the geodatabase each of these flow change points is represented as a HydroJunction on the Arc Hydro network and the HydroID of these junctions is stored as a JunctionID attribute of the corresponding HMSCode and RASCode features that are connected there (a Schematic Node for an HMS element; a Cross

Section line for a RAS cross-section). Relationships between these features and HydroJunctions are created in Arc Catalog to make them permanent.

This association of features creates the linkage between HMSCode of SchematicNode to HydroID of HydroJunction to RASCode of CrossSection that is needed to connect HMS and RAS elements under Arc Hydro.

By naming convention, the streamflow records generated by HEC-HMS are related to the hydrologic elements by means of the “B part” (see section 4.1.3) in the pathnames of the HEC-DSS cards. Thus, by populating the B part in the schematic node feature class using the HMSCode attribute it is possible to extend the relationships to the cross sections and associate the time series to them. Figure 3.16 illustrates how this association is accomplished.

Attributes of DSS1Type

DSSSID	FeatureCode	DSSFile	A Part	B Part	C Part	D Part	E Part	F Part	sUnits	sType	sDType
1	02R CO	C:\msproq\Salado_Unified\Salado_Unified.dss	02R CO	FLOW	21JAN1991	5MIN	SAL ENTIRE WS	CFS	INST-VAL	Regular Interval Time Series	
2	07R CO	C:\msproq\Salado_Unified\Salado_Unified.dss	07R CO	FLOW	21JAN1991	5MIN	SAL ENTIRE WS	CFS	INST-VAL	Regular Interval Time Series	
3	03R CO	C:\msproq\Salado_Unified\Salado_Unified.dss	03R CO	FLOW	21JAN1991	5MIN	SAL ENTIRE WS	CFS	INST-VAL	Regular Interval Time Series	
4	04R CO	C:\msproq\Salado_Unified\Salado_Unified.dss	04R CO	FLOW	21JAN1991	5MIN	SAL ENTIRE WS	CFS	INST-VAL	Regular Interval Time Series	
5	06R CO	C:\msproq\Salado_Unified\Salado_Unified.dss	06R CO	FLOW	21JAN1991	5MIN	SAL ENTIRE WS	CFS	INST-VAL	Regular Interval Time Series	
6	RC1 RU	C:\msproq\Salado_Unified\Salado_Unified.dss	RC1 RU	FLOW	21JAN1991	5MIN	SAL ENTIRE WS	CFS	INST-VAL	Regular Interval Time Series	
7	08R CO	C:\msproq\Salado_Unified\Salado_Unified.dss	08R CO	FLOW	21JAN1991	5MIN	SAL ENTIRE WS	CFS	INST-VAL	Regular Interval Time Series	
8	09R CO	C:\msproq\Salado_Unified\Salado_Unified.dss	09R CO	FLOW	21JAN1991	5MIN	SAL ENTIRE WS	CFS	INST-VAL	Regular Interval Time Series	
9	01R CO	C:\msproq\Salado_Unified\Salado_Unified.dss	01R CO	FLOW	21JAN1991	5MIN	SAL ENTIRE WS	CFS	INST-VAL	Regular Interval Time Series	
10	05R CO	C:\msproq\Salado_Unified\Salado_Unified.dss	05R CO	FLOW	21JAN1991	5MIN	SAL ENTIRE WS	CFS	INST-VAL	Regular Interval Time Series	

Record: 14 0 10 Selected Records (0 out of 10 Selected) Options

Attributes of SchematicNode

Shape*	OID	HydroID	FeatureID*	SrcType	HMSCode
Point	404	456	441	2	RC1 RU
Point	568	620	423	2	09R CO
Point	393	445	422	2	08R CO
Point	398	450	421	2	07R CO
Point	586	638	420	2	06R CO
Point	410	462	95	2	05R CO
Point	588	640	418	2	04R CO
Point	401	453	417	2	03R CO
Point	408	460	98	2	02R CO
Point	406	458	415	2	01R CO

Record: 14 0 207 Selected Records (0 out of 207 Selected)

Figure 3.16 Geographic Representation of Hydrologic Elements

3.5.6 Model Connectivity in Arc Hydro

Once the correct geographic representation of the basins to be simulated is stored in the geodatabase format, the connectivity elements can be created. This Geographically Integrated Regional Watershed Modeling System is based on the ideas shown in Figure 3.17 for model connectivity. For this integration some relationship classes need to be created and populated defining the Model Codes for each of the integrated engineering models.

Under this geographic integration, the Schematic Node feature class is populated with the HMSCodes from HEC-HMS as stored in the B part of the DSS pathname for each of the outlet nodes. In turn, the FeatureID field in the schematic nodes needs to be populated with the HydroID of the HydroJunction it represents. By means of this field association the relationship class “HydroJunctionHasSchematicNode” can be built. By doing this, the HydroJunctions now know which HMS outlet node they represent. Next, the JunctionID field in the Cross Section feature class is populated with the HydroID of the HydroJunction that is exchanging flow into a given downstream cross section. Thus, the HydroJunctions now know which flow receiving cross section it is related to. Under this spatial scheme the hydrologic elements (outlet-nodes) can be related to the hydraulic elements (cross sections).

For the flow of time series information between model elements, the streamflow records related to the hydrologic elements by means of the HMSCode (which exists in the schematic nodes) can now be associated with the cross sections. With these built-in relationships we can transfer time series information

from the geodatabase back into the DSS system but now, the time series records will be associated with the cross sections by means of the RASCode (a concatenation of stream, reach, and station IDs).

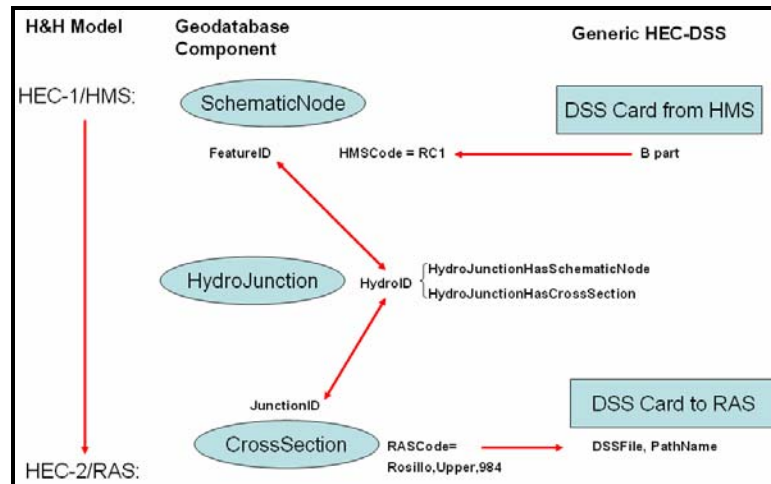


Figure 3.17 Conceptual Approach for Model Connectivity in Arc Hydro

The HydroJunction feature class in the Arc Hydro structure plays a vital role in making this geographic connectivity a reality. Its HydroID is not only used to populate foreign keys for the schematic nodes (the hydrologic side) and the Cross section (the hydraulic side) feature classes but also to contain the ModelCodes for each modeling system. Thus, the HMSCodes and corresponding RASCodes are populated on those HydroJunctions representing outlet nodes and associated with cross sections. Figure 3.18 illustrates these modeling attributes for the 10 flow exchange points in Rosillo Creek.

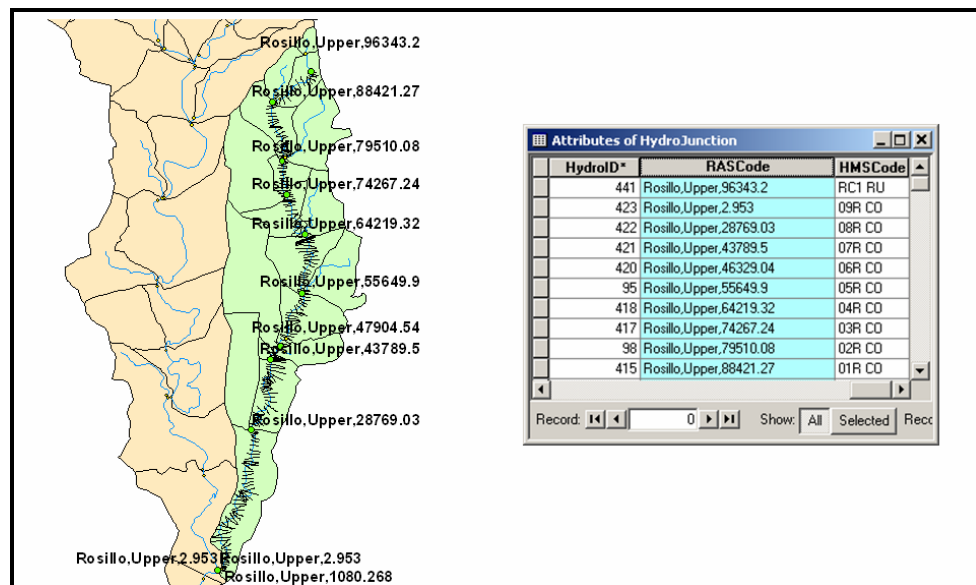


Figure 3.18 HydroJunction Features as Connectors with Hydrology and Hydraulics

To make evident and permanent the central role of the HydroJunction feature class as a bridge between the hydrologic and hydraulic models, three relationship classes are built and populated in the geodatabase data model.

- **HydroJunctionHasWatershed**, to define the outlet node for each hydro response unit in HEC-HMS
- **HydroJunctionHasSchematicNode**, to relate to a schematic network that resembles the HEC-HMS network (modeling topology).
- **HydroJunctionHasCrossSection**, to relate the cross section that inherits flow values from HEC-HMS

The above relationships around the HydroJunction feature class are shown in Figure 3.19.

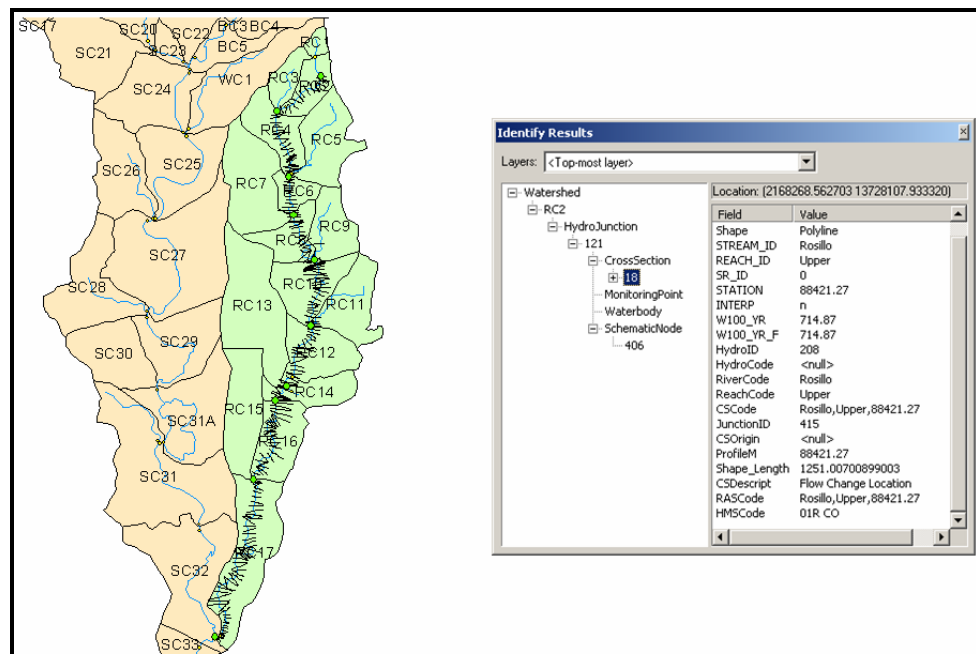


Figure 3.19 Relationship Classes for Integration Support

3.5.7 Time Series Transfer to Arc Hydro

The model connectivity is devised by a geographic association between points of information exchange. In other words, transfer of information between models is needed by clearly labeling the exchange points, and creating permanent geographic relationships between them. By doing so, the correct exchange of spatially-referenced time series can be accomplished.

An external system is needed to query the created relationship and actually do the transfer of information between the model systems. The external application should pair the HMS nodes with the RAS cross sections through existing geodatabase relationships and then transfer the hydrographs for each corresponding cross section in the hydraulic model. Given that the HEC series of

models stores its time series records in the HEC-DSS system (Data Storage System) the transfer of information has to be done by reading from DSS and writing into the Arc Hydro geodatabase.

For this project, a procedure has been developed that reads DSS files produced by HEC-HMS, identifies only the files pertaining to flow change location points, and writes them into the geodatabase. The system relies on a geodatabase structure that includes an object class containing all relevant DSS records and descriptors (DSSTSCatalog table component) and supporting Visual Basic algorithms to automate the transfer of information into this table.

The output from a given hydrologic simulation in HEC-HMS generates time series records stored in the HEC-DSS Data Storage System for time series. Within it, the DSS catalog composed of pathnames (containing block records for a single spatial location) lists all the included regular time series records, and paired values for multiple hydrologic elements. The DSS catalog also contains the flow time series records for the hydrologic nodes defined as flow change locations and for the receiving cross sections in HEC-RAS. Thus, the development of a mechanism to read the DSS catalog generated by HEC-HMS simulations and transfer the streamflow information to some intermediate “container” that can be later used to feed the hydraulic modeling system is critical. Figure 3.20 shows an extract of the DSS catalog (list of all pathnames in a HEC-DSS file) generated by the hydrologic model for the Salado Creek.

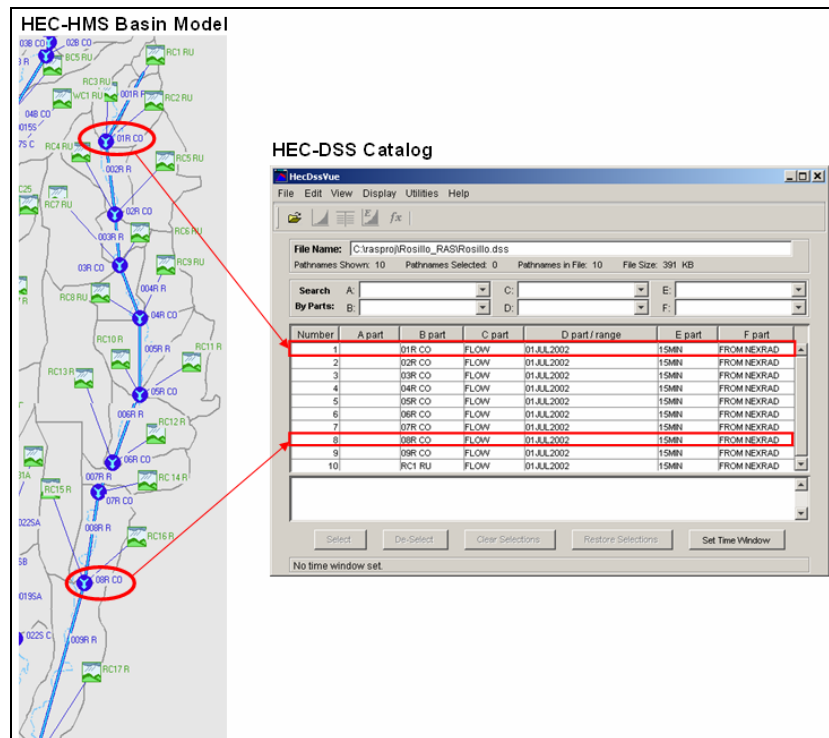


Figure 3.20 DSS Time Series content of HMS for Exchange Points

To accomplish this transfer of exchange time series information (under the proposed geographic framework), the geodatabase must contain the information (in object class or table format) to store not only the time series values but also the descriptors of them. An object class (table) for the time series values is a standard component in Arc Hydro but a new table has to be designed and included to accommodate the HEC-DSS descriptors of the time series records. This table is called the DSSTSType (DSS Time Series Type) table and contains all the needed fields for each of the time series descriptors in HEC-DSS.

DSSTSID	PathName	DSSFile	A_Part	B_Part	C_Part	D_Part	E_Part	F_Part	sUnits
1	//02R CO/FLOW/21JAN1991/5MIN/SAL ENTIRE WS/	C:\Vmsproq\Salado_Unified\Salado_Unified.dss	02R CO	FLOW	21JAN1991	5MIN	SAL ENTIRE WS	CFS	
2	//07R CO/FLOW/21JAN1991/5MIN/SAL ENTIRE WS/	C:\Vmsproq\Salado_Unified\Salado_Unified.dss	07R CO	FLOW	21JAN1991	5MIN	SAL ENTIRE WS	CFS	
3	//03R CO/FLOW/21JAN1991/5MIN/SAL ENTIRE WS/	C:\Vmsproq\Salado_Unified\Salado_Unified.dss	03R CO	FLOW	21JAN1991	5MIN	SAL ENTIRE WS	CFS	
4	//04R CO/FLOW/21JAN1991/5MIN/SAL ENTIRE WS/	C:\Vmsproq\Salado_Unified\Salado_Unified.dss	04R CO	FLOW	21JAN1991	5MIN	SAL ENTIRE WS	CFS	
5	//06R CO/FLOW/21JAN1991/5MIN/SAL ENTIRE WS/	C:\Vmsproq\Salado_Unified\Salado_Unified.dss	06R CO	FLOW	21JAN1991	5MIN	SAL ENTIRE WS	CFS	
6	//RCT RU/FLOW/21JAN1991/5MIN/SAL ENTIRE WS/	C:\Vmsproq\Salado_Unified\Salado_Unified.dss	RCT RU	FLOW	21JAN1991	5MIN	SAL ENTIRE WS	CFS	
7	//08R CO/FLOW/21JAN1991/5MIN/SAL ENTIRE WS/	C:\Vmsproq\Salado_Unified\Salado_Unified.dss	08R CO	FLOW	21JAN1991	5MIN	SAL ENTIRE WS	CFS	
8	//09R CO/FLOW/21JAN1991/5MIN/SAL ENTIRE WS/	C:\Vmsproq\Salado_Unified\Salado_Unified.dss	09R CO	FLOW	21JAN1991	5MIN	SAL ENTIRE WS	CFS	
9	//01R CO/FLOW/21JAN1991/5MIN/SAL ENTIRE WS/	C:\Vmsproq\Salado_Unified\Salado_Unified.dss	01R CO	FLOW	21JAN1991	5MIN	SAL ENTIRE WS	CFS	
10	//05R CO/FLOW/21JAN1991/5MIN/SAL ENTIRE WS/	C:\Vmsproq\Salado_Unified\Salado_Unified.dss	05R CO	FLOW	21JAN1991	5MIN	SAL ENTIRE WS	CFS	

Figure 3.21 DSSTSCatalog in Arc Hydro

After the storage system for the data was defined (the above table), some computer applications (in the form of script tools) were develop to transfer the data from the HEC-DSS system into the predefined Arc Hydro table (a so called HEC-DSS time series “bridge” application).

To accomplish this, the Arc Hydro structure must contain not only the time series records but also their corresponding type descriptors for the Arc Hydro and HEC-DSS systems. The time series records itself are stored in a standard Arc Hydro table called “TimeSeries” which contains the time stamp and its associated value. The type of time series for Arc Hydro is also stored in a standard table containing the basic descriptors of the time series: the type of variable, the measuring units, the time interval, the acquisition type, etc. In order to be able to interface with HEC-DSS and therefore with any of the HEC models, this component also contains a table storing the HEC-DSS descriptors that are needed to write into the HEC-DSS system. These three tables have associations between them and with relevant spatial features with ownership on the time series records. Figure 3.22 illustrates the conceptual design of this component.

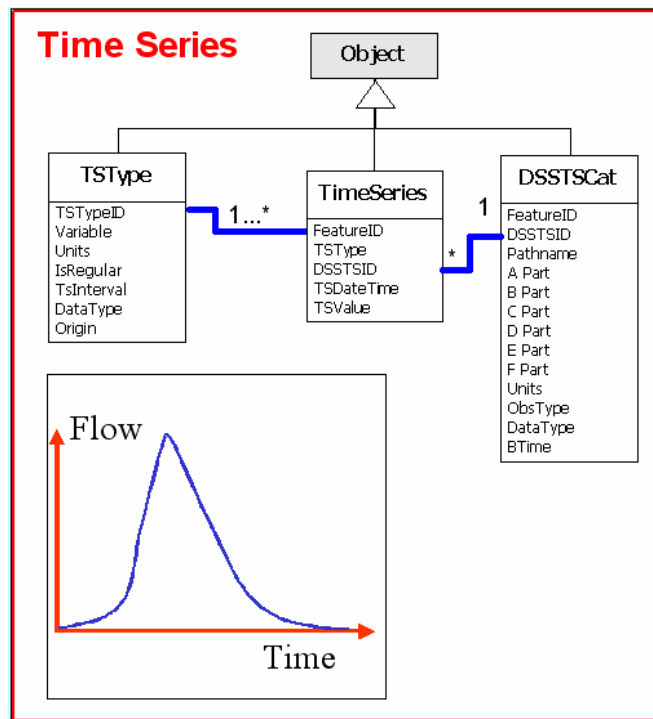


Figure 3.22 Extended Time Series Component in Arc Hydro

3.5.8 Updating the Hydraulic Model Configuration

As part of the proposed integrated framework, updates of the hydraulic model configuration need to be done at run time to reflect and register the new flow inputs is the main purpose of this phase. Extraction of data from the Interface Data Model for HEC-RAS containing the geodatabase representation of the hydraulic model and formatting it to generate required input files to run the HEC-RAS model drawing from the newly generated input data in HMS.

Once a geodatabase containing all the necessary elements to achieve connectivity is built, the next step is to update the hydraulic model configuration by generating the DSS file for HEC-RAS with the transferred flow values from

HEC-HMS, creating the corresponding flow file in which the exchange cross sections are now referenced to the transferred DSS records and its peak flow value (for steady state setups), creating a new plan file for HEC-RAS that includes a pointer to the new flow file, and updating the project file with pointers to the new flow and plan files if not existing.

By having model-centric elements inside the geodatabase it is possible to create model specific files with their content. After the geographic association between model elements is done, the DSS time series records from HEC-HMS must be used to create a flow file for the HEC-RAS setup. The DSSTSType table with the DSS descriptors for the transferred streamflow data and the Cross Section feature class with the correct HEC-RAS model code are used as primary sources for the new Flow file. The Figure 3.22 shows the details of the information being read and transferred from the geodatabase to the HEC-RAS flow file.

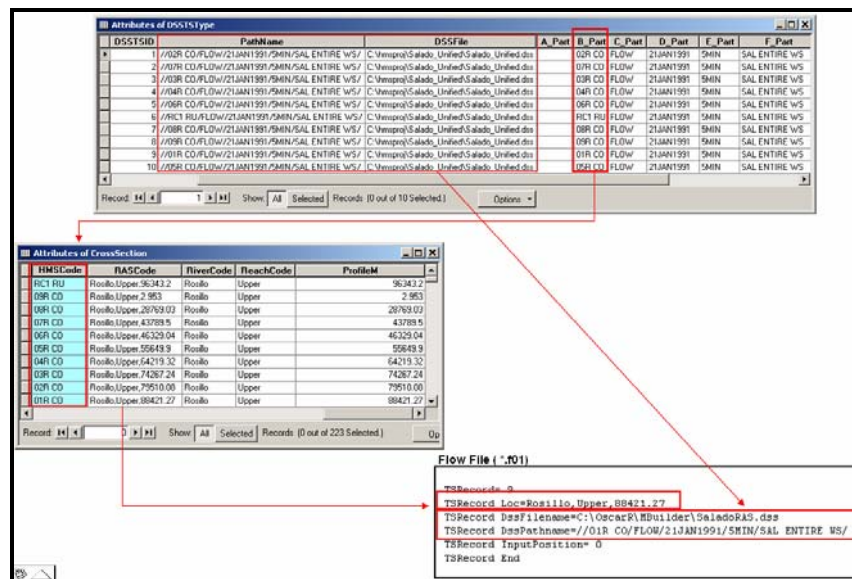


Figure 3.23 Creating a Flow File from geodatabase content

The Steady State Flow File for HEC-RAS has two basic components. The flow peak values for each cross section feature and for each flow event and the boundary condition for each river reach and each flow event. Figure 3.23 shows the constructed flow file using information stored in the geodatabase for the hydraulic model of Rosillo Creek.

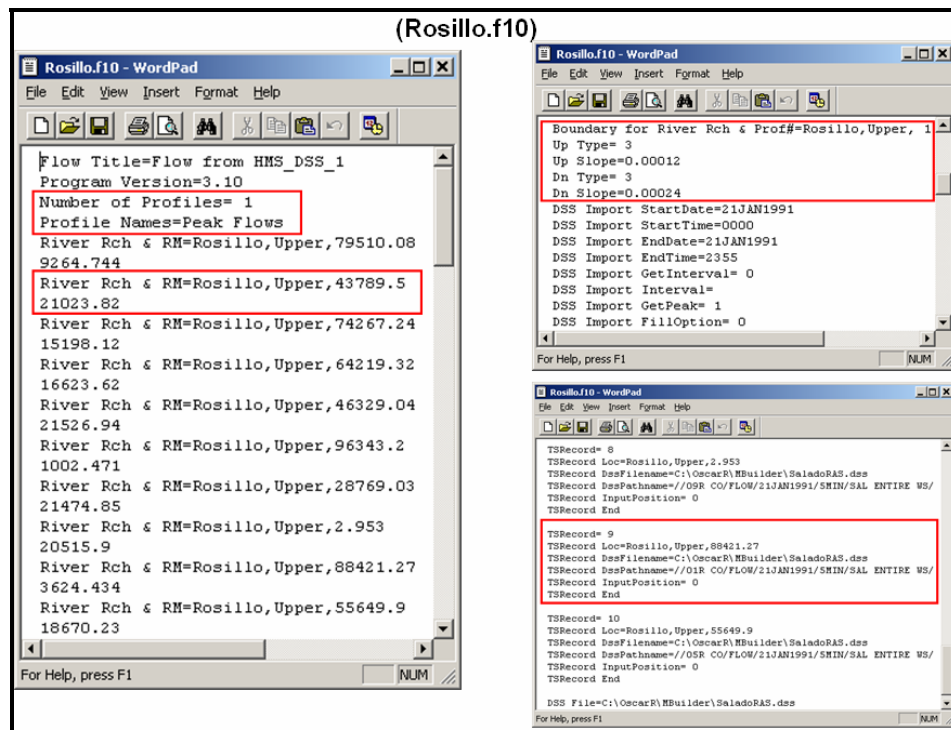


Figure 3.24 Components of a Steady State Flow File

This task is accomplished by a procedure that writes all the files needed for HEC-RAS to do its flow calculations (dss, flow, plan, and project files). The currently developed tools include only the steady state flow configurations but the system might be extended to include unsteady state flow configurations (Snead, 2000) by pointing to the DSS time series file and pathname sources. For the steady state flow configuration, the peak flow values from the HMS generated hydrographs are stored in DSS and extracted by the application so that the Flow file above shown can be entirely created.

The time series component of this model integration was accomplished by developing two computer applications to execute time series transfer operations

between the HEC-DSS file system and the geodatabase system. The conceptual framework for this transfer procedure is depicted below using each of the graphical user interfaces and input-output files using icon-type representations. First, the process reads the target geodatabase and source DSS file to transfer the information over the time series tables inside the target geodatabase. Second, the process uses an application to transfer the flow time series records needed to be exchanged back into a HEC-DSS file for the HEC-RAS system and generates and updates the corresponding flow, project, and plan file to obtain a new HEC-RAS configuration.

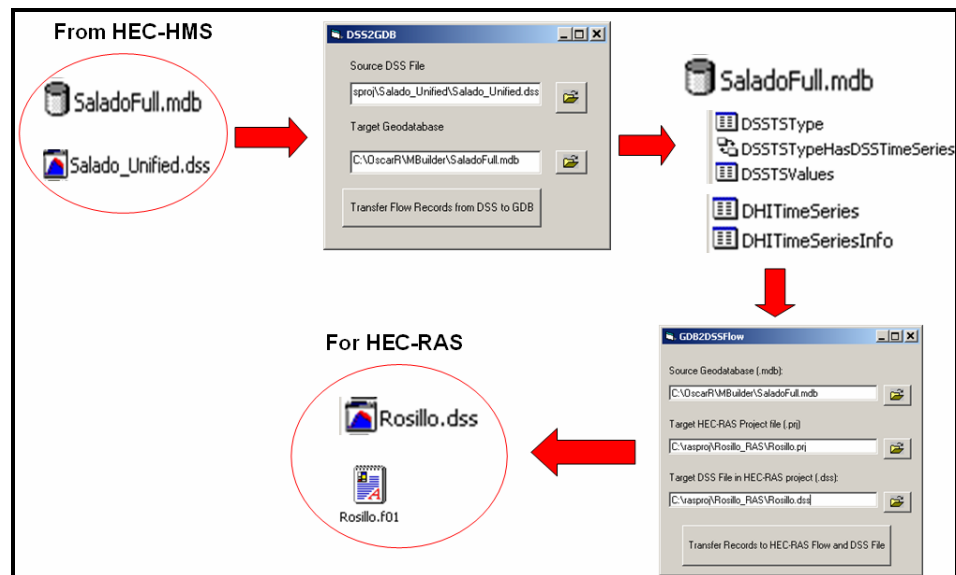


Figure 3.25 Time Series Transfer Schematic

The time series transfer procedure outlined above represents a two-step process that works but can be impractical for “near real time operations”. An alternative to the time series transfer application procedure is to have the time

series transfer tasks incorporated into a sequence of events (model workflow) that can be run automatically in a single-step.

The current research contributions corresponding to the linkage operations described above have been used in a computer application that generates a floodplain map from a NEXRAD map of precipitation, the “map to map” use case.

3.5.9 Automated Floodplain Delineation

The hydraulic model simulations produce water surface elevations at each cross section defined in the model configuration. To enable floodplain impact analysis capabilities the extent of the floodplain affected by a given hydrologic event must be mapped. Diverse floodplain delineation schemes have been developed with different levels of sophistication and improvements are constantly being formulated (Noman et al., 2001). To configure an integrated and dynamic system, the inclusion of the floodplain delineation component to the workflow is central. The floodplain delineation procedure involves the following main conceptual steps:

- Generation of a water surface TIN based on water surface elevations obtained for each cross section
- Conversion of the water surface TIN into a water surface raster
- Subtraction of the land surface raster from the water surface raster to obtain the water depth raster
- Conversion of the water depth raster into a flood inundation polygon by selecting only positive depth grid values and dissolving the polygons

The more specific steps for the post-processing phase in charge of the flood map delineation are shown and have been incorporated into the Map2Map workflow by means of standard geoprocesses in Arc GIS 9 and customized script tools to achieve the post-processing steps needed for flood inundation mapping.

- The HEC-RAS ASCII export SDF file is converted to XML format for GIS assimilation
- The Elevations in XML format are associated with cross sections in Arc Hydro
- The elevations along the cross sections are used for interpolation and generation of a water surface TIN
- The water surface TIN is clipped to the floodplain boundary
- The water surface TIN is converted to raster format
- The water surface raster is intersected with the terrain raster to produce a water depth raster
- The water depth raster is converted to polygon format
- The water depth polygon is dissolved to generate the flood inundation map

Figure 3.25 illustrates the schematic sequence for the above floodplain delineation steps

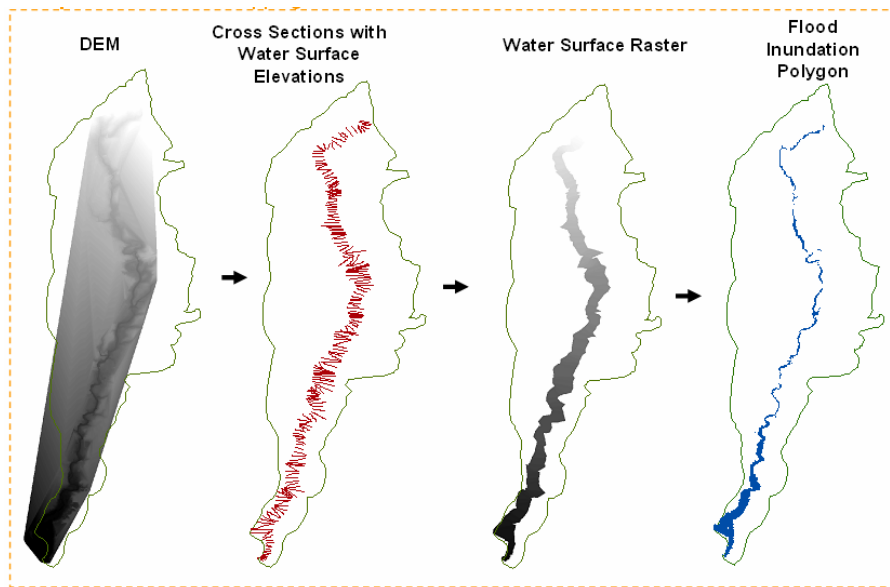


Figure 3.26 Floodplain Delineation Schematic

Chapter 4 Map2Map Components

4.1 MODELING COMPONENTS FOR THE HISTORIC USE CASE

This chapter provides a more detailed description of all the processing tasks involved in the integration. These groups of tasks are designed to run in a sequential workflow in which each process's output becomes the input for the next process in line. Each of the described processes might be a customized script tool that executes a task that is out of the GIS realm, a standard geoprocess already included in current GIS software, or submodels that in turn may contain customized or standard processes.

The ModelBuilder schematic for the Map2Map workflow is shown in Figure 4.1 for the use case of historical precipitation events. This workflow starts at the upper left process “Radar2GDB” (first yellow rectangle) and ends at the final right process “Create Flood Polygon from Surfaces” (last process rectangle).

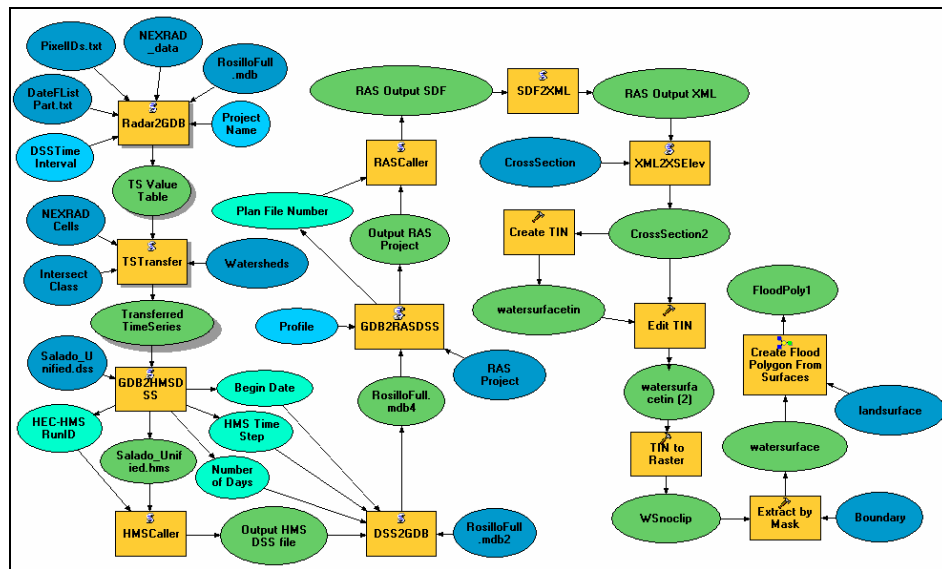


Figure 4.1 Map2Map Workflow in ModelBuilder for Historic Event Use Case

4.1.1 The Radar2GDB Script Tool

The first step in the Map2Map application (first yellow box above) deals with the access and transfer of precipitation time series data (to drive the hydrologic simulation in HEC-HMS) from the available NEXRAD format and into an extended Arc Hydro format for time series. For the current implementation of the Map2Map application, the rainfall data is provided by the network of NEXRAD radars along the U.S. The mappable format of this data corresponds to the rectangular grid termed Hydrologic Rainfall Analysis Project (HRAP) grid having a nominal grid spacing of 4 km x 4 km. Each of the rectangular grid cells is identified by a PixelID with which a time series of rainfall data is associated.

A subset of NEXRAD time series data for Texas was provided by the Guadalupe Blanco River Authority in Texas as a series of ASCII files. The subset was comprised of 552 hourly text files for a storm event in late August of 2002 and for the entire state of Texas. Thus, for the current format being used, the rainfall data are stored in individual ASCII data files named after the hourly time stamp they contain.

A snippet of the file content of one of the text files is shown below for a filename “0701200204.” The filename contains the time stamp embedded in it with the date format MMDDYYYYHH. Thus, the filename “0701200204” contains a rainfall data for an event occurring on July 01, 2002 at hour 04. Figure 3.4 shows a snippet for this data text file. The text file contains a header defining the first columns as a pixel id (P_ID) and the second column as the precipitation value in inches (PRECIP_IN). At the end of the data records, the text file ends with an END statement.

The precipitation values on the above Figure are associated with a PixelID pointing to a NEXRAD cell in the mappable HRAP format. The layer containing the HydroResponse feature class with the NEXRAD cell definition was provided by the Brazos River Authority in Texas. This layer is a statewide NEXRAD HRAP grid in which the “Rfcthsn_ID” field in the grid layer corresponds to the P_ID field (first column Pixel ID) in the above data text file.

Given that each of the hourly text files of precipitation contains data for all of the NEXRAD cells in Texas (50,150 cells), the NEXRAD cells covering the specific study area must be extracted and a new NEXRAD layer for the watershed

under analysis must be created. In addition to creating a NEXRAD layer with the pixels of the case study at hand, the application needs to select the precipitation values associated only to the NEXRAD cells covering the study area. For this purpose, the Radar2GDB script tool uses a PixelID text file (with the list of needed NEXRAD cell IDs) to identify from the text files those NEXRAD cells that are specific to the study area under analysis and extract their related precipitation values.

The Radar2GDB script tool also needs to know the time window of the storm event to be simulated so that out of the multiple hourly files describing the storm event the proper text files are read. To define this time simulation window a list of file names with the desired time stamps is provided to the application as a DateFileList text file (DateFList.txt). An example of this input file is shown in Figure 4.3 that spans from July 1, 2002 at 4 AM up to July 1, 2002 at 10 PM.

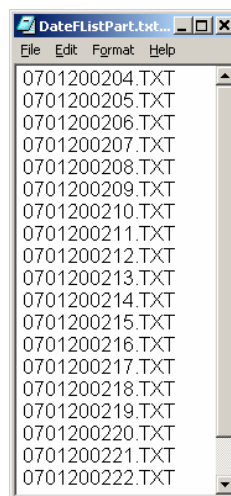


Figure 4.2 List of Date Filenames defining a Simulation Time Window

A folder location is also provided to the application to identify the target folder location in which the precipitation ASCII data files are located. The application then matches available cells/pixels with NEXRAD polygon features in the Arc Hydro geodatabase, and imports the time series of rainfall into standard Arc Hydro TimeSeries tables (e.g., TimeSeries and TSType) and an additional table (DSSTSType) for storing HEC-DSS descriptors associated with the Arc Hydro descriptors. All the three tables (TimeSeries, TSType, and DSSTSType) must already exist in the geodatabase (as part of the schema design); along with all the fields associated with these tables in the Arc Hydro definition and in the HEC-DSS time series data system.

After implementing the tool, the TimeSeries table is populated in such fashion that the FeatureID for each timestamp value is populated with the PixelID of the NEXRAD cell it belongs to. Figure 4.3 illustrates the 15 NEXRAD cells covering the Rosillo Creek and a section of the TimeSeries table showing the first time stamp precipitation values associated with each NEXRAD cell by means of the FeatureID (also shown as a backdrop is the watershed segmentation of Rosillo Creek).

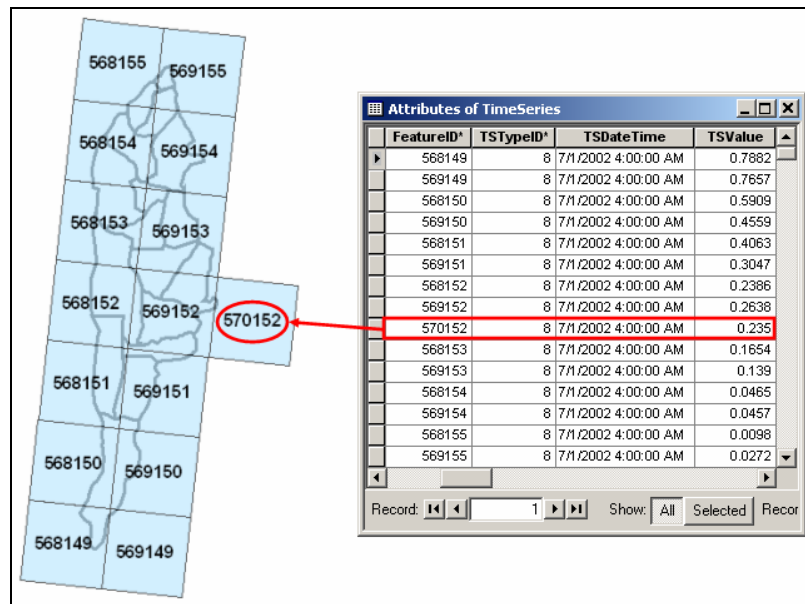


Figure 4.3 Time Series values associated with NEXRAD cells in Arc Hydro

In summary, the Radar2GDB script tool requires the folder location of the NEXRAD data, the list of pixel ids covering the study area, a list of date-filenames defining the desired simulation time window, and the target Arc Hydro geodatabase to receive the precipitation time series. The output from this tool corresponds to the Arc Hydro TimeSeries table holding the actual precipitation records for each time stamp inside the simulation time window. The ModelBuilder schematic for this tool is shown below in Figure 4.4.

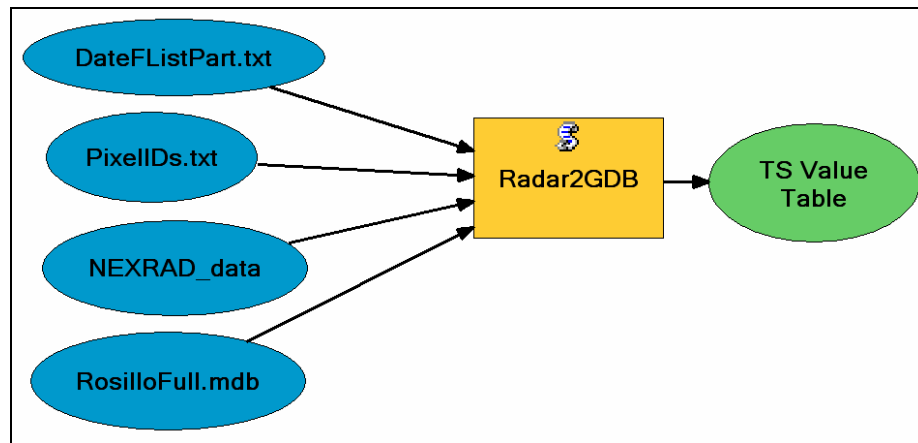


Figure 4.4 Radar2GDB Model Component

4.1.2 The Time Series Transfer Script Tool

The Time Series Transfer script tool transfers time series associated with the NEXRAD cells/pixels onto the Watersheds (Subbasins) of interest as defined in the hydrologic model being used (the HEC-HMS setup). By doing this, the input rainfall hyetographs are associated with each watershed in the basin as required by the hydrologic model. The tool performs the transfer operation by analyzing the extent to which each NEXRAD cell covers a given Watershed, and then based on that areal extent; relative weights for each intersecting subbasin area are assigned to compute a weighted average of rainfall for that subbasin at each time step (areally weighted precipitation values). Figure 4.5 shows the areal disaggregation of subbasin RC6 RU based on covering NEXRAD cells (568154, 569154, 568153, and 569153) to calculate the precipitation values for the subbasin.

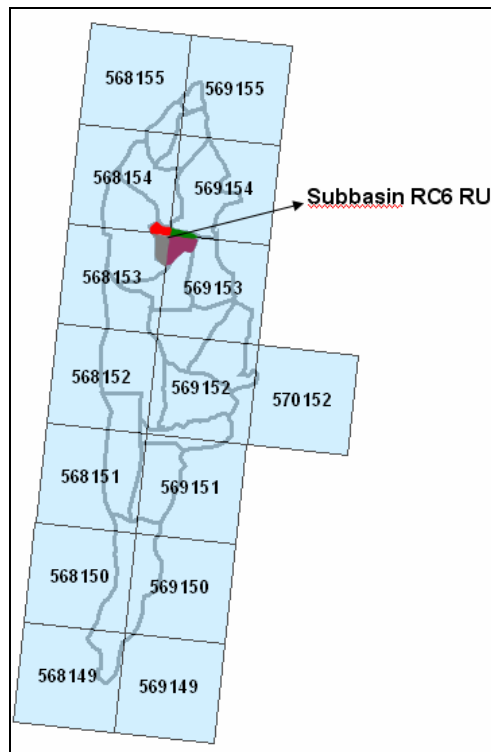


Figure 4.5 Subbasin Partition based on NEXRAD Cell Overlaps

For this tool to work a feature class created by intersecting the Watersheds with the NEXRAD cell polygons must exist as it is used to compare the overlapping areas. This feature class may be created by using the Intersect function from the Geoprocessing Wizard in Arc GIS 8.x or using the new and corresponding geoprocessing tools in Arc GIS 9, available through the standard ArcMap user interface or Arc Tool box respectively. The output from the Radar2GDB tool is an output table of transferred time series data. The output time series may be appended to existing records in the output table, or the existing records in the table may be deleted before adding the newly created time series

records. Figure 4.6 illustrates the conceptual process that is executed by the Time Series Transfer tool.

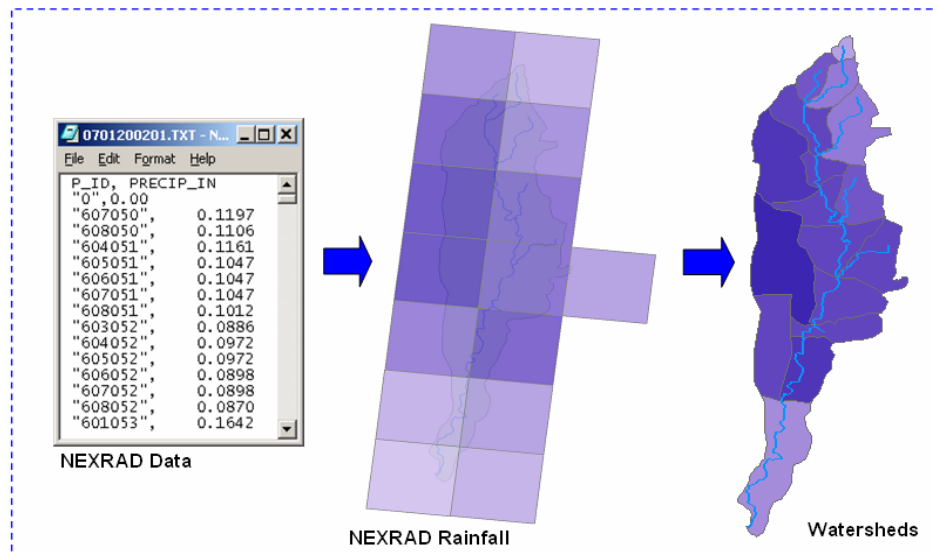


Figure 4.6 Associating NEXRAD Data with Watersheds

Based on the above discussion, the “Time Series Transfer” tool requires the filename and location of the NEXRAD cells polygon layer (normally stored under the HydroResponseUnit feature class in Arc Hydro), the filename and location of the layer containing the basin subdivisions stored under the Watershed feature class in Arc Hydro, the layer corresponding to the intersection of the previous two layers (NEXRAD cells and Watersheds), and the filename and location of the time series table storing the precipitation values associated with the NEXRAD cells. The output from this process (green oval) corresponds to the filename and location of the created time series table with records now associated

with the watersheds. Figure 4.7 shows the Model builder schematic for this process.

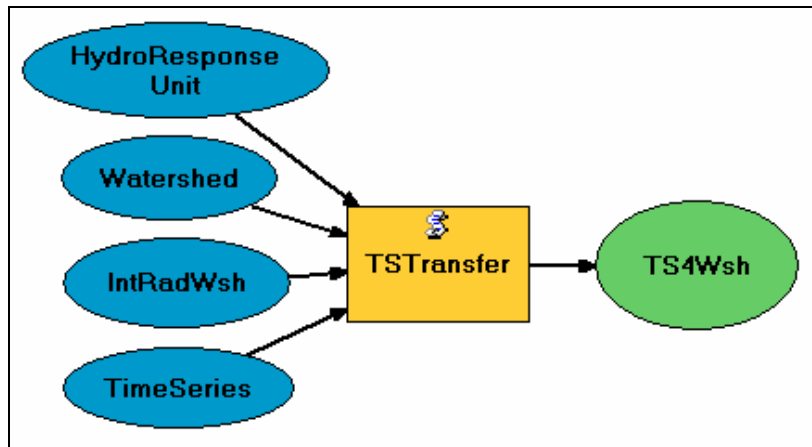


Figure 4.7 TSTransfer Model Component

After the tool is executed the output time series table (TS4Wsh in this case study) is populated with the same time series records stored in the original TimeSeries table but now the Feature ID is populated with the HydroID of the watershed that has ownership over the data. Figure 4.8 illustrates some sections of the new time series tables and their associations to related subbasins.

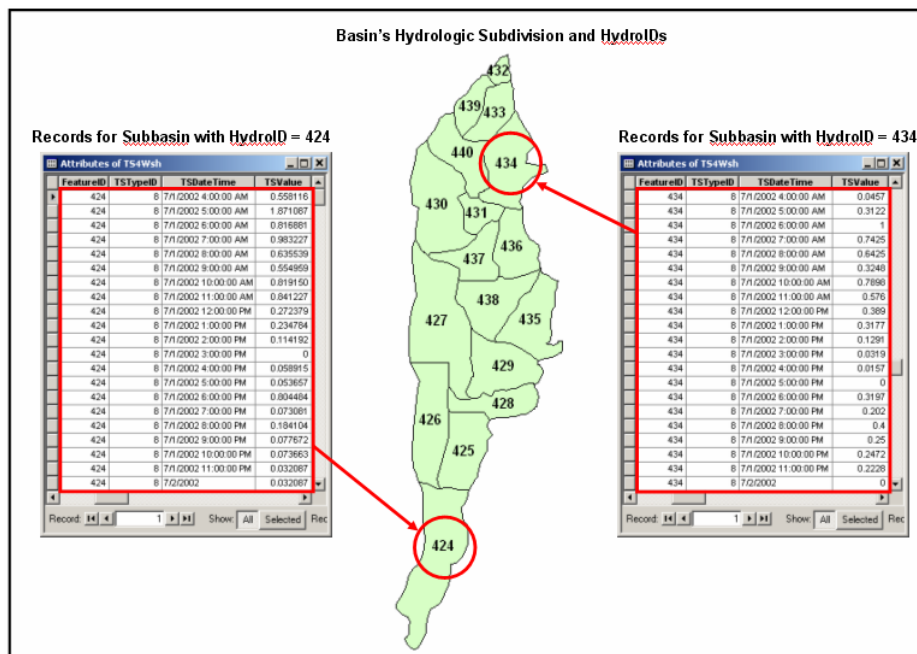


Figure 4.8 Time Series Tables and Related Subbasins

4.1.3 The GDB2HMSDSS Script Tool

The HEC-DSS Data Storage System is the database system developed by the Hydrologic Engineering Center for efficiently storing and retrieving water resources data including regular and irregular interval time series data, paired data (curves), and text (alphanumeric) data. Its development was motivated by the idea of having a more efficient system for water resources based on blocks of contiguous data elements that are uniquely labeled for faster retrieval and storage. Even though the HEC-DSS system can store almost any type of data, it is most efficient at storing large blocks of data (e.g., time-series data). These blocks of time series data are internally stored as records in HEC-DSS, and each record is given a unique name called a "pathname." The HEC application programs have

the ability to read from and write to the HEC-DSS database. This capability facilitates the use of observed data by the models as well as the passing of information between HEC software programs.

HEC-DSS was written in FORTRAN 77 (whose development started in 1979 prompted by the Kissimmee River study in Florida) and designed to be called by FORTRAN programs. Today the use of the DSS system has expanded into real-time data storage applications that allow rapid storage and retrieval of files containing 100,000 records or more. The DSS is made up of a set of utility programs that allow access (interface) by multiple programs. That is, DSS makes use of several subroutines in the software library HECLIB that enable other programs to interface with it. At its creation, back in 1979, the interfacing implementations were centered on HEC applications programs like the well-known Flood Hydrograph Package (HEC-1) and the Expected Annual Damage (EAD) program. At that time, interfacing implementations with external systems outside of the HEC series of models were not envisioned, much less interfacing implementations with external programs developed in programming languages other than FORTRAN.

The original motivation behind the HEC-DSS system has some similarity with the motivation of the present work, that is, the need to avoid the tedious step-wise approach in which most studies are performed. Under this step-wise approach, the data is passed from one analysis program to another in a manual fashion favoring a time consuming and error-prone workflow that is functional but not very productive nor efficient. Process analysis using the same type of data

having different formats and that are sequentially related can take advantage of a unique DSS data format and a library of subroutines to empower a seamless end-to-end integration like the one described herein with GIS as the gluing platform.

Therefore, the current integration scheme herein described relies on this library for the data exchange necessary for wiring together the external modeling components involved in an automated floodplain delineation system.

Thus, the library of subroutines that make up the DSS system empower it with the ability to readily be used with other applications programs to enable external retrieval and storage of information. The previous concept of having methods (subroutines) in charge of doing specialized DSS tasks was a milestone for interfacing abilities that 20 plus years later can be exploited by interfacing implementations with various external systems including the present work.

Data within a DSS file is stored in blocks (or records) and each record is identified by a unique identifier called a pathname. Besides the data itself, information about the data (metadata) is stored in the pathname parts and also in an internal header containing the data type, units of the data, and so on. In this way, through information contained in the pathname and the internal header, the data is self-documented and no additional information is required to describe it. By having access to the pathname parts and the header, any external system may have access to the DSS records which implies that the information can be recognized and understood even years after it was stored.

As mentioned before, a major component of the HEC-DSS database is the structure of the pathname (the unique identifier label for each block of DSS data).

The pathname may consist of up to 391 characters. It is, by convention, separated into six parts with a forward slash delimiter. Again, when accessing a block of data in a DSS file, the full data's pathname (including optional parts if they were defined at creation) must be given along with the header's metadata, that is, beginning time, units of the data, and type of data stored in the internal header of the data's block. By convention, each part in the pathname is referenced by the letters A, B, C, D, E, and F, which are delimited by a forward slash with the form: **/A/B/C/D/E/F/**.

The pathname is used to describe the data in enough detail that various application programs can write to and read data from HEC-DSS by “simply” knowing the pathname (USACE, 1995). A brief description of each pathname part follows to provide the fundamentals for the integration of time series as crucial component in Map2Map.

Part A: A logical grouping name for all records (blocks) that belong to a specific project or study (e.g., the name of the watershed case study). This part is optional.

Part B: Identifier of the geographic element that owns the time series records (site location of the data). For example the gage ID that provides a georeferenced location to the data. This part is not optional.

Part C: Variable type description (e.g., FLOW, PRECIP, STAGE, etc.). The user can define his own variable type descriptions or use the recommended C-part names as listed in the user's guide (USACE, 1995). This part is not optional.

Part E: Time Interval of stored data. It is a combination of an integer number with an alphanumeric time interval. Valid data intervals are: 1MIN, 2MIN, 3MIN, 4MIN, 5MIN, 10MIN, 15MIN, 20MIN, 30MIN, 1HOUR, 2HOUR, 3HOUR, 4HOUR, 6HOUR, 8HOUR, 12HOUR, 1DAY, 1WEEK, 1MON, and 1YEAR. This time interval indirectly defines the DSS block length and in turn the format of the valid Part D as described next.

Part D: Block start date or starting date of the data block. It must be a 9-character military style date for the start of the standard data block (not necessarily the start of the first piece of data). Regardless of when the first valid data occurred, the block starting date is a constant standard value that depends on the block's length. This part is not optional. The length of the block determines the format of the starting date as shown below:

If the block length is a century the starting date must be the beginning of that century: 01JAN1900 for a block length covering the 20th century (regardless of when the first valid data occurred). Used for yearly data intervals.

If the block length is a decade, the starting date must be the beginning of that decade: 01JAN1960 for a block length covering the 1960's. Used for weekly and monthly data intervals.

If the block length is a year, the starting date must be the beginning of that year: 01JAN1982 for a block length covering the 1982 year. Used for daily data intervals.

If the block length is a month, the starting date must be the beginning of that month: 01MAR1982 for a block length covering March. Used for 15MIN,

20MIN, 30MIN, 1HOUR, 2HOUR, 3HOUR, 4HOUR, 6HOUR, 8HOUR, and 12HOUR data intervals.

If the block length is a day, the starting date must be the beginning of that day: 24MAR1982 for a block length covering March 24. Used for 1MIN, 2MIN, 3MIN, 4MIN, 5MIN, and 10MIN data intervals.

In practice, the time interval (Part E) defines the block length and in turn, the block length defines the format of the valid starting date (Part D). That is, Part E determines the format of Part D. This means that for a 10MIN dataset covering a period of 6 days there are 6 blocks of data (6 pathnames in DSS catalog). To some extent the HEC-DSS system might be considered a “time series block model” and not a truly sequential time series model.

Part F: This part relates to a unique descriptor of the data’s origin. Examples of this part might be the simulated scenario or plan that generated the data, the name of the model run that generated the values, or the nature of the data (observed, simulated, and recorded, etc.).

Taking advantage of the library and consideration for all the issues aforementioned, an application (GDB2HMSDSS script tool) was developed in the Visual Basic 6.0 programming language and included as part of the Map2Map workflow. This tool also relies on the HEC Library of FORTRAN functions and subroutines for time series data exchange (heclib50.dll, i.e., heclib version 5.0) containing all the functions needed for access to the HEC-DSS database system. This heclib50.dll is actually the same library used by the HEC-HMS and HEC-RAS models to read and write time series and curves from and into DSS at

execution time and for their corresponding simulations and archival purposes. All the DSS functions and subroutines included in the library are well documented in its programmer's manual (USACE, 1991).

4.1.3.1 Accessing NEXRAD precipitation data

The GDB2HMSDSS script tool is in charge of transferring rainfall time series data associated with each Watershed/Subbasin from the geodatabase to an HEC-DSS file that is referenced by the HEC-HMS model. Each schematic node in the geodatabase matches a subbasin or junction element in the HMS basin file, identified by the HMSCode attribute field. This tool is called prior to calling HEC-HMS, so that the HMS project will have access to the latest rainfall data generated from the geographically integrated framework. In addition to adding the rainfall time series to the HEC-DSS binary file, this tool also updates the HEC-HMS gage and control files accordingly to register the new storm event and its corresponding time simulation window. This sets up a complete HEC-HMS model configuration in which the new time series of rainfall contained in the HEC-DSS file will be appropriately used to drive the hydrologic simulation.

After the radar precipitation records have been stored in GIS and associated with the watersheds (subbasins in HEC-HMS nomenclature) by means of the previous Time Series Transfer process, they need to be transferred from the geodatabase time series tables to a HEC-DSS binary file to make them available for simulations by the HEC-HMS modeling system. The data exchange of precipitation records for each subbasin involves not only the time series values themselves but also the corresponding descriptors for those datasets. As explained

before (in section 3.4.7), the HEC-DSS descriptors are being stored inside the Arc Hydro data model by means of an object class (non-feature table) that extends its core content, the DSSTSType table. The structure of this table has been designed with care and consideration for all the time series attributes that are needed not only for the time series data exchange but also for the integration scheme itself (see Figure 3.21). On the other hand, the bulk of the time series records are stored in the standard time series Arc Hydro table containing the timestamp and value definitions for each record in the series.

The application (GDB2HMSDSS script tool) references tables in the Arc Hydro geodatabase to get the HEC-DSS descriptors (from the DSSTSType), the Time Series Values (from the Arc Hydro TimeSeries table), and the Watershed table from which the number of features/rows is used to define the number of watersheds that own precipitation records which need to be read and transferred to DSS.

To achieve the final goal of transferring the rainfall dataset into a HEC-DSS file, various subroutines from the HEC library need to be called for the data exchange to take place. The ZOPEN, ZSRTS, and the ZCLOSE DSS software subroutines are the key subroutines implemented in this application.

After the GDB2HMSDSS script tool reads the Arc Hydro Time Series tables from the geodatabase, it writes the records to HEC-DSS (by accessing the data exchange library of utilities) to make them available for the user-specified hyetograph meteorologic option of HEC-HMS. The following section elaborates

on the steps needed to transfer the data and update the hydrologic configuration to reflect the new input setup.

4.1.3.2 Accessing the HEC-DSS System

The first thing done by the application is to open the HEC-DSS file that stores the precipitation records for the new HEC-HMS input configuration. The ZOPEN DSS subroutine is in charge of attaching a DSS file to a program (the GDBToHMSDSS application in this case) and obtains basic information from the file, allowing reading and writing to proceed. In other words, ZOPEN is the first DSS subroutine to be called when interfacing with the HEC-DSS system. The DSS file does not need to exist before the ZOPEN subroutine is called and it will be created if it is not present. By doing this, the application provides a generic functionality for data storage into a new or existing DSS file.

The main FORTRAN subroutine that writes regular time series into HEC-DSS, the ZSRTS subroutine (Store Regular-Interval Time Series Data), has been incorporated into the main program (GDB2HMSDSS) as a public module that is invoked every time it is required. The main input and output arguments that are passed through this utility are described bellow denoting the descriptors needed to store datasets into the HEC-DSS system. This description is of particular relevance for the definition of all the parameters that have to be present in the GIS system before the data exchange operation takes place as shown bellow.

ZSRTS(IFLTAB(1), sPath, Len(sPath), sBDate, Len(sBDate), sBTime, Len(sBTime), INVals, fValues(1), sUnits, Len(sUnits), sType, Len(sType), IPLAN, ISTAT)

IFLTAB (I/O): The DSS work space used to manage the DSS file. This is the same array used in the ZOPEN call. It must be dimensioned (1 To 600) as a long integer and only the first element, IFLTAB(1), has to be passed to the subroutine.

sPATH (I): The pathname of the data to be stored. The maximum length of a HEC-DSS pathname is 80 characters, with each pathname part not exceeding 32 characters. If passed, the D part of the pathname is ignored, as it is superceded internally by the specific arguments defining the data's time window: beginning date, beginning time, and the number of values to store.

Len_sPath (I): The length of the pathname, sPath, programmatically set as "Length (sPath)."

sBDate (I): The beginning date of the records' time window (date of first value). It must have a valid HECLIB date format. Although a variety of date formats can be recognized for the date, the recommended format is DDMMYYYY, that is, 05Mar1968. Other valid entries for dates are: 05Mar68, 5 March 1968, march 5, 1968.

Len_sBDate (I): The length of the starting date text string, sBDate. Programmatically set as "Length (sBDate)."

sBTime (I): The beginning time of the time window. This must be a standard 24-hour clock time (e.g., "1630"). Any time-offset is implied by the date and time specified (for example, if daily data is measured at 9:00 a.m., then setting the beginning time to "0900" implies an offset of 9 hours. A colon separating the hours and minutes is optional (09:00).

Len_sBTime (I): The length of the starting time programmatically set as “Length (sBTime).”

INVals (I): The number of values to be stored in HEC-DSS. It is used to define the end of the time window (ending date and time) covered by the record set.

fValues (I): The data values to be stored. This array of values must be in a sequential order with no gaps (regular time series), and with the first value having the beginning date and time.

sUnits (I): The units of the data (e.g., CFS, CMS, FEET, etc.).

Len_sUnits (I): The length of sUnits, programmatically set as “Length (sUnits).”

sType (I): The type of the data (e.g., "PER-AVER"). Acceptable data types and units are shown below:

- PER-CUM: Total precipitation occurring in each time interval with units “mm” or “in”. Used for incremental precipitation.
- PER-AVER: Average flow during each time interval with units “cms” or “cfs”. Used for time intervals equal to or greater than 1 day.
- INST-CUM: Instantaneous cumulative precipitation at the end of each time interval with units “mm” or “in”. Used for precipitation mass curve.

- INST-VAL: Instantaneous flow at the end of each time interval with units “cms” or “cfs”. Used for time intervals less than 24 hours.

Len_sType (I): The length of sType. Programmatically set as “Length (sType).

IPLAN (I): An argument to indicate whether or not to write over existing data.

0: Always write over existing data

1: Only replace missing data flags in the record (DSS flag for missing values is -901.0)

4: If an input value is missing (-901.0), do not allow it to replace a non-missing value in existing record (keep the existing known value)

ISTAT (O): A status parameter indicating the success of the writing operation. Its possible values are:

0: The data was successfully stored.

4: all of the inputs provided were missing data flags (-901.0).

10: A fatal error occurred.

11: The number of values to store (NVALS) is less than one.

12: Unrecognized time interval.

15: The starting date or time is invalid.

24: The pathname given does not meet the regular-interval time series conventions.

51: Unrecognized data compression scheme. Valid schemes are 0 to 5.

52: Invalid precision exponent specified for the delta data compression method. The precision exponent range is -6 to 6.

Based on the previous parameter definition for ZSRTS, the application needs to extract and generate HEC-DSS Element descriptors for each subbasin's hyetograph from the existing DSSTSType table. This table contains all the needed arguments (pathname and headings) for the data exchange process performed by the functions inside the heclib50.dll library in charge of the HEC-DSS import/export of time series and paired data datasets (that is, ZOPEN, ZSRTS, ZCLOSE).

Once the application extracts all the HEC-DSS descriptors from the DSSTSType table for each of the watersheds of the study area it places them in a 2-dimensional array in which the first dimension represents the number of the watershed and the second dimension contains the specific DSS descriptor (pathname parts) being stored (Figure 4.9).

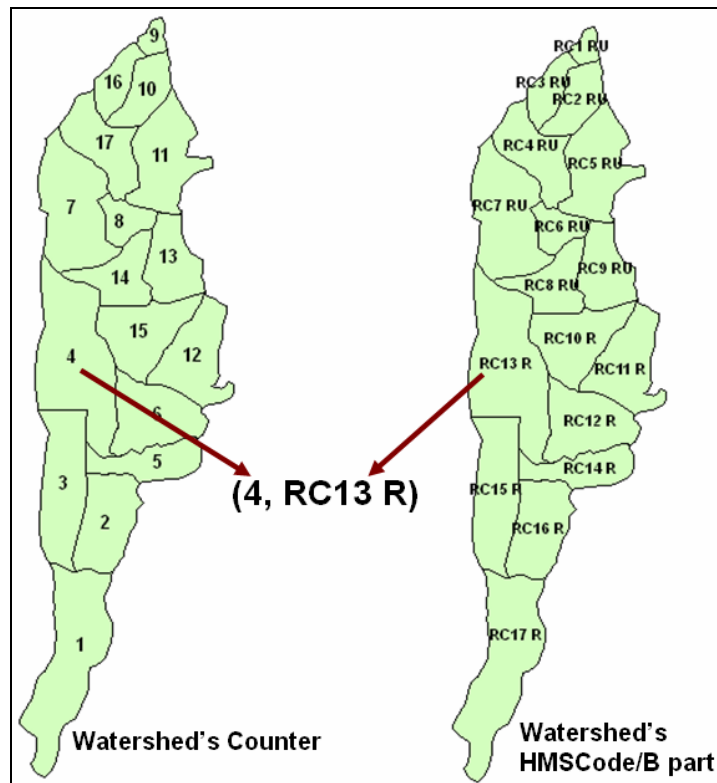


Figure 4.9 2-D Array storing watershed's counter and HEC-DSS descriptors

4.1.3.3 Generating the Meteorologic File

Communication of data with HEC-HMS must be done via ASCII files. The interfacing program must write the data needed by the external program into input files, using the information read or generated by the previous process in the workflow.

The meteorologic model in HEC-HMS performs computations of precipitation and evapotranspiration on each subbasin in the subbasin list. This component is defined by the meteorologic file in which various setup alternatives can be chosen. Precipitation for each subbasin can be obtained by several

methods: user hyetograph, user gage weighting, inverse-distance gage weighting, gridded precipitation, frequency storm, standard project storm, and the SCS hypothetical storm. The first four methods are used for the analysis of historical precipitation datasets and the last three methods are designed for producing synthetic precipitation events. The user specified hyetograph method may also be used for feeding a synthetic storm computed outside the HEC-HMS model. The user has then four options for defining historical events and four options for defining synthetic events.

At first, the gridded precipitation option seems to be the logical choice for the ingestion of the precipitation records since the source of precipitation is NEXRAD data and that the radar values are estimated and remapped to a national square grid called the Hydrologic Analysis Project (HRAP) grid (Reed and Maidment, 1999). Under this grid-based option, quasi-distributed cell-based hydrologic attributes can be used for the ModClark basin transform method in the HEC-HMS meteorologic definition.

Even though the gridded precipitation method was devised for using radar rainfall data inside HEC-HMS, for the Map2Map implementation the gridded precipitation option was not used for a couple of reasons:

- In HEC-HMS, gridded data must be at 1-hour intervals and the time interval in the control specifications must also be set to 1 hour. This poses a limitation on the resolution of the NEXRAD data that the application might ingest (specially for real-time applications with small time interval of 15 minutes or less) and a limitation on the

resolution of the output runoff hydrographs that may miss the flow peaks and accurate volume amounts.

- To preserve the original HEC-HMS model configuration as it was provided by the local agencies in San Antonio. In fact, most of the hydrologic model configurations are developed with the user-specified hyetograph option and simulations from them should be kept as close as possible to the original setup.

In light of the aforementioned reasons, the user defined hyetograph option, as selected in the original configuration of the HMS model was employed to assign the rainfall hyetographs to each subbasin as the first alternative to be implemented for the Map2Map application.

The user-specified hyetograph option allows the user to enter/define the depth and temporal distribution of a historical or hypothetical storm event. In practice, in HEC-HMS the rainfall values for each subbasin are entered and interpreted as if it were rainfall at a gage. The model, however, applies that “point rainfall measurement” over the entire subbasin for the hydrologic simulations (a lumped approach).

For most of the options available to ingest rainfall into the HMS model, the final goal is to have a rainfall hyetograph assigned to each subbasin element as defined in the segmentation of the basin under study. The gridded precipitation option on the other hand assigns a hyetograph to each grid-cell and the model performs transformation functions at the grid-cell level also.

Once the hyetograph is defined (for each subbasin in this case) it can be entered into the HEC-HMS system manually (internal gage definition) or by making a reference to it through a HEC-DSS pathname record inside the flow file content (external DSS gage definition). The previous statement implies that a meteorologic file needs to be generated referencing the appropriate DSS records containing the rainfall hyetograph for each subbasin.

4.1.3.4 Writing the Gage File

After the HEC-DSS descriptors have been extracted from the geodatabase table and the target HEC-DSS filename and directory location is established, the application writes the gage file (.gage) for HEC-HMS based on the subbasin's names and using the extracted HEC-DSS pathnames (one per subbasin) and directory location. The main purpose of this file is to store a complete list of all type of gages inside or outside the basin that could potentially be used by the HEC-HMS model setup. Some or all of the precipitation gages for example are used to define the meteorologic input of the model and depending on the type of meteorologic model to be used; the gages will need to be georeferenced by means of geographic coordinates (the inverse distance weighting method in which relative position is needed). For the Map2Map implementation only the HEC-DSS filename and location with the specific DSS card pathname pointing to the rainfall time series is needed.

4.1.3.5 Writing the Meteorologic File

By means of the DSSTSType table the application (GDB2HMSDSS script tool) gets hold of the DSS descriptors needed for the exchange and the HMS input

files can then be written based on those DSS definitions. The first ASCII input file written by the application is the Meteorologic file (.met file extension) which contains the definition for the rainfall hyetograph for each modeled subbasin. The most relevant components of this file (from the integration perspective) are the pointer to the HEC-DSS file location and the subbasin names.

The reference to the HEC-DSS file location (in the meteorologic file) is assumed by the application to be at the same directory location of the HEC-HMS project being integrated. This directory contains the initial configuration of the HEC-HMS model for the corresponding case study (Rosillo Creek in this case). Thus, the output HEC-DSS file with the rainfall datasets defining a hyetograph for each subbasin is located in the same folder location as the HMS project.

Next, each subbasin is referenced to one of the registered gages containing the rainfall hyetograph for it. To accomplish this, the related gage name is registered in the meteorologic file under the subbasin section owning the hyetograph. By doing this, the subbasin will point to a single gage (a virtual gage at a centroid subbasin location) that contains the rainfall hyetograph to be hydrologically routed through that subbasin. Therefore, the meteorologic file contains not only a registry of all the subbasins that comprise the system but also the hypothetical gage that has ownership on the affecting rainfall hyetograph for each subbasin. By doing this, the meteorologic file is updated by the application with the information of the new rainfall event that wants to be transformed as recorded by the NEXRAD radar.

The resulting runoff hydrograph is owned by the hydrologic element (junction, subbasin, reservoir or source) topologically upstream of the reach receiving the runoff hydrograph to be superimposed in the sequential routing process. Most of the time (as in the current case study) the hydrologic element owning the hydrograph (flow change location point) will be the junction representing a confluence point of a tributary and for the most upstream reach, the hydrograph will belong to the headwater subbasin itself (an upstream boundary condition). That is, the runoff generating element will remain as the owner since it is also the most upstream element of the receiving reach.

To systematically establish the ownership of the resulting runoff hydrograph by the most upstream element of the receiving reach, the B part of the HEC-DSS pathname is used to store the name of the hydrologic element through which the hydrograph is entering the system (a main stem reach). In short, the B part will be populated with the name of the element that passes the hydrograph to the next downstream reach (which sometimes happens to be the very same element that generates the hydrograph or most usually a junction representing a confluence point).

4.1.3.6 Extracting the Time Stamp and Value

Next, the application (GDB2HMSDSS script tool) opens the Arc Hydro TimeSeries table to get the time series data for each Watershed based on the unique combination of the FeatureID of each Watershed and the variable type being read (precipitation in this case). The precipitation values for each subbasin are then placed in a 2-dimensional array in which the first dimension represents

the number of the watershed and the second dimension contains the index number of the precipitation value being stored (a counter for rainfall values in the hyetograph).

4.1.3.7 Writing the Control File

Next, the application writes the Control File for HEC-HMS which defines the simulation time window (simulation period). The control file contains the starting and ending dates and times for the simulation time window as well as the simulation time interval (having a complete number of time intervals inside the simulation time window).

In order to completely setup a HEC-HMS model, the control file (.control) must be generated. This file contains the starting date, starting time, ending date, and ending time for the runoff simulation. For historical storm events the starting date and time is given by the beginning of the storm event and for hypothetical events, the starting date and time is arbitrarily set by the analyst. The definition of the ending date and time for the entire simulation (not just the duration of the storm event), however, must be carefully selected to define an appropriate and “hydrologically” complete simulation time window (from starting time to ending time). The final goal is to make the computation time horizon to go long enough to completely describe the rise and fall of the hydrograph being generated as a result of the rainfall-runoff transformation (rising and recession limb definitions). That is, the time it takes for the flood wave to go from its base flow before the beginning of the storm event to its peak flow and then back to its base flow.

Thus, the time period for the entire simulation must be large enough to encompass the full range of streamflow magnitudes that may occur during the storm event, from the baseflow values to the peak flow values and back to the baseflow values to capture the entire rise and recession of the hydrograph's shape.

One approximation for this is to define a simulation time window that encompasses the entire rainfall event duration plus the hydrologic response time of the entire watershed plus the hydraulic response time of the channel system. This total duration is then the storm duration plus the travel time of the watershed up to the outlet. The storm duration can be read from the time series realization of the event, but in principle the travel time of the watershed has to be selected with care and consideration for the intensity of the storm, the initial moisture conditions, the level of development of the watershed, etc. Thus, in practice, the total simulation time window is obtained by incrementing the starting date and time of the simulated storm event with the storm duration plus a reasonable estimate for the travel time of water within the watershed.

The simulation time interval (computational time step) determines the resolution of the model results computed during a run. The entered gage data is linearly interpolated to the selected time interval (from hourly gage data to 15 minute time steps in this case study). The analyst must define at what time interval the computed flow hydrographs will be written to HEC-DSS. In general, this time interval should be selected to give an adequate number of points to fully define the shape of the computed hydrographs (rise and fall) without losing key information about the flow peaks or volume of the hydrographs. To make sure

these rough criteria are honored, some people recommend the use of a computation interval that is equal to or less than the time of rise of the hydrograph divided by 24. This way if the flood wave goes from its base flow to its peak flow in 24 hours, then the computation interval should be equal to or less than 1 hour (USACE, 2000a).

Once the target HEC-DSS file has been defined (named and located at same HMS project's location), the simulation run has been given a name (RunID), the beginning date and time has been retrieved from the geodatabase DSSTSType table, and most importantly, the time step, ending date, and ending time have been estimated based on needed hydrograph resolution and full description of its shape (considering event and travel time durations) respectively, a new control file can now be written for the desired storm event.

4.1.3.8 Registering a New RUNID Element

The next step in the application is in charge of registering the new combination of basin file, meteorologic file, and control file in the existing RUN file. This application has considerations for already existing registries with the same combinations of basin-met-control file names so that the new run combination is only added to the RUN file if it does not exist in the current RUN file.

Finally, the application invokes the ZSRTS from the library of FORTRAN utilities inside the heclib50 DLL to store the regular time series of precipitation into the HEC-DSS file. This file will have the same name of the HEC-HMS project filename and will be located at the same location.

After the ZSRTS subroutine is executed, the application invokes ZCLOSE another DSS software subroutine that updates and releases a DSS file from a program. This closing subroutine is quite important to make the DSS file available to other programs like HEC-HMS itself (the next process in the workflow). Figure 4.10 shows the schematic of the GDB2HMSDSS script tool.

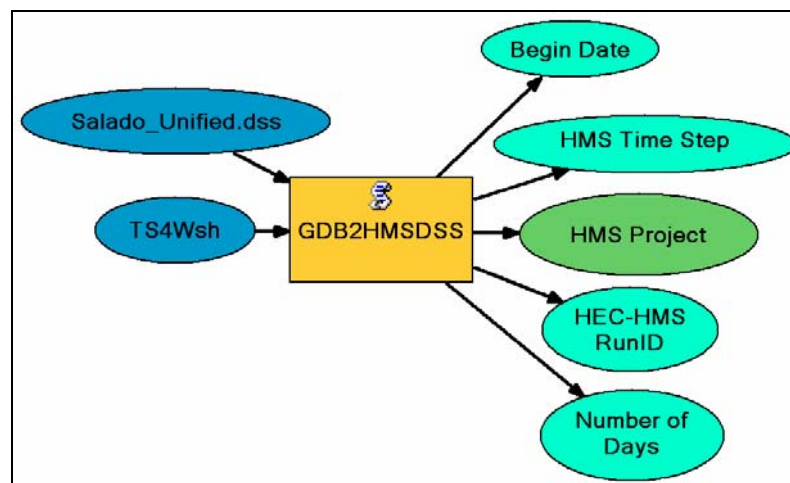


Figure 4.10 GDB2HMSDSS Model Component

The time series transfer operation is supported by the DSSTSCatalog table which contains all the HEC-DSS descriptors needed to accomplish the transfer back to HEC-DSS by means of the DSS subroutine ZSRTS. The schematic of the transfer operation is shown in Figure 4.11 in terms of the Arc Hydro table that holds the DSS catalog information and the receiving HEC-DSS binary file.

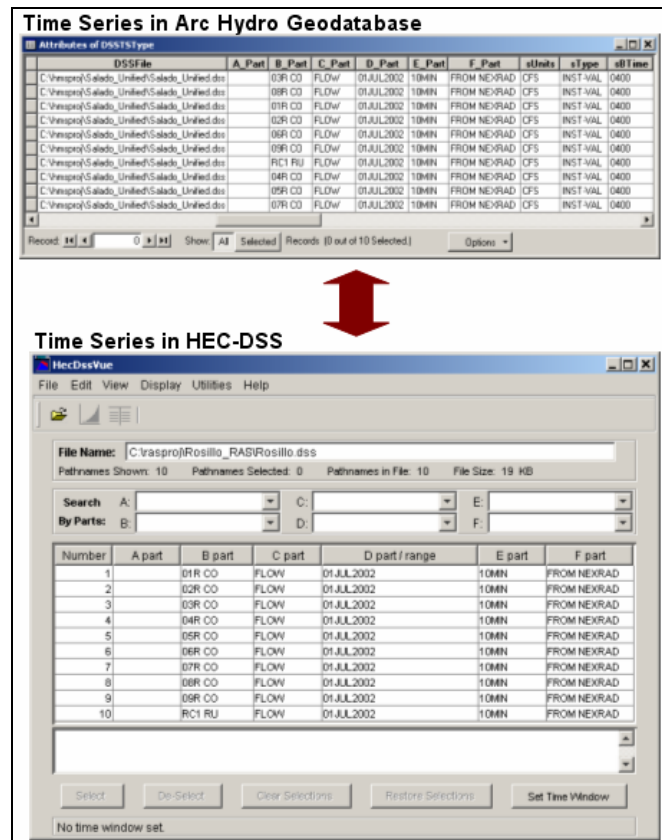


Figure 4.11 Time Series Transfer between Arc Hydro and HEC-DSS

4.1.4 The HEC-HMS Caller Script Tool

The HMSCaller script tool calls HEC-HMS to perform rainfall-runoff transformations for each Watershed in the basin (Figure 4.12). The result is a set of runoff hydrographs for predefined nodes in the stream network, with the nodes representing Watershed outlets and stream confluences, streamflow gages, etc. The output hydrographs are stored in HEC-DSS as regular time series records with properly defined descriptors.

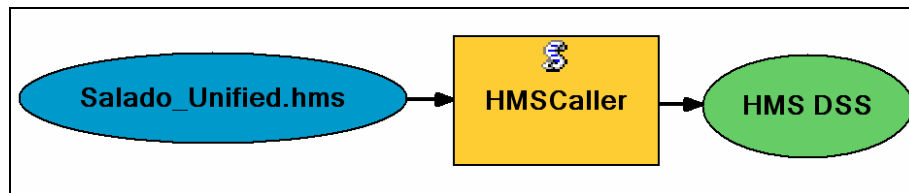


Figure 4.12 HMSCaller Model Component

Map2Map requires the launching and monitoring of external programs to execute a hydrologic simulation and to determine if the launched application has terminated and fully completed its output generation before continuing processing and executing the next line of code or the next process in line.

The ability to externally call and execute an existing HEC-HMS model is critical for Map2Map. DOS-based programs have traditionally been called by Command-Prompt statements that include the name of the application, some type of execution mode index (switches), and the required arguments representing the input and processing options. To execute DOS-based programs from an external application, programmers normally use special programming functions (e.g., the Shell function in Visual Basic) in charge of launching the external application.

For the Map2Map application we are interested in calling an external program from a VB DLL that can be incorporated in as a script tool inside Model Builder. Obviously, if the standard ESRI geoprocessing tools (hammer icons) cannot do the required task (a complete hydrologic simulation), an external program needs to be invoked performed a job that is not inherent to GIS functionalities.

When HEC-HMS is called, it runs asynchronously from ArcGIS. Therefore, the HMSCaller DLL must have functionalities to pause execution of the geoprocessing workflow until HEC-HMS completely finishes execution and before returning control to ArcGIS.

All external applications launched by Shell functions run asynchronously, which means that the launching application is not notified when the launched application (also known as the shelled application) has finished execution. It implies that eventually, the next program statements in the launching application (next tool in model builder's workflow) will continue execution before the launched application (HEC-HMS in this case) has completed execution.

For the proposed integration to succeed, it is required that the launching applications or processes run in a synchronous fashion in which every process in line keeps track of the execution of the previous process and waits for it to be completed before the next process starts executing. The ability to wait is particularly critical for this integration because under the current integration scheme, the output of one process becomes the input for the next process inline. Therefore, starting a process with an incomplete input configuration will generate an error and will halt the execution of the entire workflow or will propagate errors along the integrated components of the workflow. Thus, the integration scheme needs to start an application using the Shell tools and wait for the shelled program to end before continuing processing and executing the next line of code.

To avoid asynchronous behavior, several options can be implemented. For the present work, the Shell function was enhanced with delay capabilities that are

based on a process handle (a programmatic reference to the application currently executing, a handle on the process id). Once created, the process handle is used to wait for the running object to finish execution. In practice, to wait for the external program to terminate, the program may use the WaitForSingleObject API call or the Windows Scripting Host (WSH).

The programmatic options for calling an external program from Visual Basic are briefly described below. As mentioned before, for Visual Basic 6, Shell command is the most recommended way for interfacing with an external application (if it does not provide an object library to access its functionality). In its general syntax the code statement returns a process object id that is stored in a long integer variable. The Shell command then contains the pathname of the file to be executed and the desired style of window for the execution. An example of the Shell statement in Visual Basic follows.

```
lngPID = Shell("c:\...\MyTextFile.txt", vbNormalFocus)
```

To wait for the external program to terminate, the program might use the WaitForSingleObject API. The WaitForSingleObject uses a process handle generated with the previous process id of the open process (the launched external program).

For Visual Basic.NET, the external call operation is done through the System.Diagnostics namespace which has a Process class that is used to launch external programs. The program needs to pass in the name of an executable file or a filename with an extension associated with an executable application. The general syntax is shown bellow:

```
System.Diagnostics.Process.Start("c:\\...\\MyTextFile.txt")
```

To wait for the launched process to end, the program must invoke the `Process.WaitForExit` method. By doing this, the application will stop executing until the launched process exits.

After the execution is completed, the system continues execution and the required output is now available for the next process in line. This enhanced module that implements the `Shell` function for launching the external program and the `WaitForSingleObject` API for waiting for the launched program to terminate execution was named the “`ShellAndWait`” module and its implementation was most effective at providing a synchronous behavior that avoids errors generated from incomplete sequentially generated input. In this case for making sure the `Map2Map` application waits for the complete execution of `HEC-HMS` before the next script tool is executed.

The current version of `HEC-HMS` does not expose its functionalities through an object library in which basic methods for executing the entire model, executing an individual hydrologic process, importing input, or exporting output can be instantiated from an external application (as opposed to `HEC-RAS`, see section 4.1.10).

However, the Hydrologic Engineering Center has made it possible to execute `HEC-HMS` from a script file (programming code stored as text to quickly perform simple tasks) that contains simple command line statements for opening a `HEC-HMS` project, computing a simulation, and closing and exiting the program.

These simple command line statements need to be executed from within a script file that in turn is executed by a DOS-type command line.

The DOS-based command-prompt statements for opening, computing, and exiting a HEC-HMS project have the following syntax:

```
OpenProject("MyHMSProject", "C:\hmsproj\MyHMSProject")
```

```
Compute("MyRunID")
```

```
HMSExit(1)
```

The first line opens the HEC-HMS project by means of two arguments, the project's name (name of the .hms project file) and the path directory location of the HMS project. In most cases, the default installation of HEC-HMS creates the C:\hmsproj\ folder location as the default location for storing all the HEC-HMS projects.

The second line statement executes one of the RunID configurations of the previously opened HEC-HMS project. A HEC-HMS project may have multiple RunID configurations representing unique combinations of the Basin, Meteorologic, and Control components of the project. Thus, besides knowing the main name of the HEC-HMS project, the calling application needs to know which specific RunID should be executed from the many internally stored run setups.

The third line statement is in charge of exiting the HEC-HMS program after the execution is completed. This exiting command line has a numeric argument denoting the selected exit option.

These DOS-based command line statements for opening, computing, and exiting HMS need to be executed and automated in a batch mode. Besides having

to be executed together and in sequence these statements also need to be executed by a single command line that references the three statements to the HEC-HMS program. Therefore, to accomplish the batch execution of the three statements, a script file must be created that contains the three lines for the specific HEC-HMS project that needs to be executed and can be called by a HEC-HMS DOS-based command line statement.

So, in short, HEC-HMS can be executed from a DOS-based command line statement and one of the options for the statement is to execute the program based on a sequence of command line statements stored in a script file. The syntax of the command line statement is shown below.

```
hms -s MyScriptFile
```

The `-s` command line option refers to the script file option. It means that the three lines for opening, computing, and exiting will be stored in a Python script file. In other words, to execute the three batch lines from a Python script the `hms` command-prompt function needs to be executed as a DOS-based command line. The execution of the three lines is then solved by generating the Python script file at run time with the appropriate syntax and argument values.

The question now is how to execute the `hms` statement that runs the script file. To accomplish this, a batch file can be generated to store the `hms` statement only and execute it through the already discussed `Shell` function in Visual Basic. The `Shell` function in Visual Basic was enhanced with the “`WaitForSingleObject`” API as discussed above to avoid the asynchronous behavior and allowing HEC-HMS to fully finish before control is returned to the next process in line.

To programmatically accomplish this, the HMSCaller Application first generates the Python text file containing the three command lines for opening, computing, and exiting HEC-HMS. This file is generated with the same filename and at the same folder location as the HMS project. After the Python file is created, HMSCaller generates the batch file containing the command line for executing HEC-HMS using the Python file as the passing argument defining the name and location of the HMS project file to be executed.

Once the batch and the Python file have been created the application uses the `ShellAndWait` module (Shell command enhanced with the `WaitForSingleObject` function) to externally execute and monitor HEC-HMS. The Shell command is then used to execute the batch file that executes HMS using the Python file to contain the specific name and location of the HEC-HMS project to be executed.

4.1.5 The DSS to GDB Script Tool

In general, the DSS2GDB script tool is responsible for transferring time series data from an HEC-DSS file to a geodatabase time series table. This tool is run after HEC-HMS has completed its simulation, and for the purpose of bringing the runoff hydrograph time series into the geodatabase. The model builder component describing this application is shown in Figure 4.13.

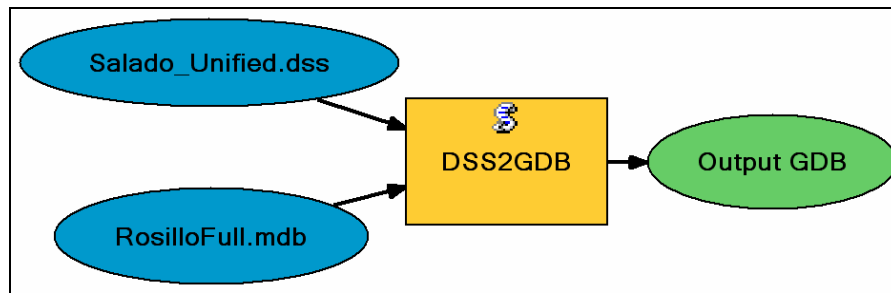


Figure 4.13 DSS2GDB Model Component

Locations of flow data change points are usually defined at the upstream end of each reach where flow source points are defined by tributaries entering an additional flow hydrograph to the main river. This is so because the reach definition is done by having in mind all the existing confluence points (tributaries entering the main river channel) and all the existing divergence points (flow split points). It is possible however to define a flow change location inside a reach if the reach segmentation missed a source point.

The hydrologic simulation in HEC-HMS is responsible for providing the runoff hydrographs for all the flow change location points that import streamflow into the main channel subject to hydraulic analysis. In general, every single reach location receiving flow (the most upstream end and all the tributary confluence points) from subbasins inside the watershed under study must be treated as flow change location points in HEC-RAS and the entering flow must be defined through the steady flow data component. By doing this, the hydraulic routing will be properly affected by those flow additions.

Since the runoff hydrographs generated by the output of HEC-HMS are stored in a HEC-DSS file, the flow values must somehow be extracted from DSS

to provide the hydraulic input for flow data definition in HEC-RAS. Given the fact that both programs have the ability to read from HEC-DSS (they both are HEC models empowered with DSS functionalities), it might be possible to reference and read the flow data directly from the HEC-HMS developed DSS file. In other words, since the two models (HMS and RAS) read DSS files the hydrologic DSS file can directly be used to provide input for the hydraulic setup. In practice however a “technical” issue arises that impedes taking advantage of the all-embracing DSS compliance. The hydrologic simulation in HEC-HMS might generate runoff hydrographs that are stored in HEC-DSS with multiple pathnames inside the DSS catalog as opposed to a single pathname that can uniquely be referenced by the HEC-RAS flow data configuration. This situation usually happens when a time series requires several blocks due to the “D part” definition in HEC-DSS.

To avoid those situations in which HMS generates multiple DSS pathname entries for a single runoff hydrograph at subbasin outlets, the Map2Map application extracts the needed runoff hydrographs from HEC-DSS (stored in multiple pathnames or not) and transfers them to the project’s geodatabase. Once the hydrographs are stored in the geodatabase they can later be extracted from the geodatabase to generate a single and non-ambiguous DSS record containing a given runoff hydrograph that can be properly referenced by HEC-RAS. This fact also contributes to keeping the integrity of the HEC-RAS configuration by defining a HEC-DSS file inside the RAS project folder location having information only relevant for the hydraulic simulation.

The above discussion represents the central justification for the DSS2GDB script tool for transferring the runoff hydrograph into the geodatabase. The extraction of the needed hydrographs from the HMS DSS file requires searching through the entire DSS catalog for the specific hydrographs pertaining only to the hydraulic flow change locations. The DSS2GDB application searches the DSS catalog using a filtering criterion based on unique combinations of pathname parts to properly identify all parts of the required runoff hydrograph.

4.1.6 The Geographic Component

Each point modeling element in HEC-HMS (subbasin, reservoir, junction, diversion, source, and sink) will have in DSS a runoff hydrograph attached to it describing the dynamic hydrologic routing for the simulated rainfall event. This hydrograph, stored in HEC-DSS, is associated with the specific hydrologic element by means of the name of the hydrologic element which is by convention stored in the B part of the corresponding pathname. So, by using the B part time series information any specific element can be found. A given element however may have multiple types of records (like a subbasin might have precipitation records and flow records associated with it in DSS). This means that extracting the flow records for a given element based solely on the B part is not enough. To precisely locate the runoff hydrograph of any given element, the search criteria must also filter by: variable type (C Part), beginning date (D Part), time interval (E Part), and by the optional global identifier (A Part). By doing so, the application locates the needed pathname inside the catalog and extracts the

information into the extended Arc Hydro geodatabase (TimeSeries, TSType, DSSTSType tables).

Before the previous filter search can be accomplished, the application needs to know the value of the main search criterion, the B part or name of the element that owns the data. This corresponds to any hydrologic element providing a flow hydrograph for the reach that needs to be hydraulically routed in HEC-RAS. From the hydrologic schematic in Figure 4.14, there are 10 flow change location points for the reach. The upstream elements from the reach segments that provided a runoff hydrograph are 1 subbasin and 9 junctions (confluence points). These hydrologic elements have a corresponding geographic representation in Arc Hydro through the Watershed and Hydrojunction feature classes.

The network topology in HEC-HMS describing the connectivity between the involved hydrologic elements can also be reproduced in Arc Hydro by means of an existing functionality, the Schematic Network. Therefore, Arc Hydro can be used to hold a geographic representation of the hydrologic network that can be used for the proposed geographically integrated modeling system. The Schematic Network generated off Hydrojunctions and Watersheds will provide GIS with the same network topology inside HEC-HMS and to empower the geographic integration of the modeling systems. The Hydro Network is considered the backbone of Arc Hydro. This geometric network is defines the topological connection of HydroEdges and HydroJunctions. All types of features can eventually be connected to the Network besides the points and lines that make up the HydroJunction and HydroEdge feature classes. The way other features can be

connected to the network is by means of relationship classes with HydroJunctions representing outlets or instream point representations. That is, any point, line, or area feature can be connected to the network through a HydroJunction relationship.

The schematic network in Arc GIS is comprised of Schematic Nodes and Schematic Links representing area and point features that are topologically connected. In this case, the Schematic Network in Arc Hydro also resembles the topologic network connections in HEC-HMS. The Schematic Nodes in particular represent all the possible HMS hydrologic elements (subbasin, reservoir, junction, diversion, source, and sink) but now inside a geographic framework. All the schematic nodes (representing HMS elements) get automatically connected to the HydroJunction by assigning the HydroID of the HydroJunction at the HMS node location to the FeatureID attribute of the Schematic Node feature class.

Shape*	OID	SrcType	HydroID	FeatureID*	HMSCode	FType
Point	15	2	15	441	RC1 RU	Flow Change Location
Point	29	2	29	423	09R CD	Flow Change Location
Point	4	2	4	422	08R CD	Flow Change Location
Point	9	2	9	421	07R CD	Flow Change Location
Point	30	2	30	420	06R CD	Flow Change Location
Point	21	2	21	95	05R CD	Flow Change Location
Point	32	2	32	418	04R CD	Flow Change Location
Point	12	2	12	417	03R CD	Flow Change Location
Point	19	2	19	98	02R CD	Flow Change Location
Point	17	2	17	415	01R CD	Flow Change Location

Figure 4.14 Arc Hydro and HMS connectivity through SchematicNode Relationship

The HydroJunction, used for connecting features in Arc Hydro, is also used as the central geographic integrator between HEC-HMS and HEC-RAS in Map2Map. In other words, the HydroJunction (Arc Hydro's main network connector) will reach out for the hydrologic elements by connecting the Schematic Nodes (HEC-HMS hydrologic elements) by means of a relationship class (HydroJunctionHasSchematicNode) that uses the HydroID of the HydroJunction as the primary key and the FeatureID in the Schematic Node as the foreign key.

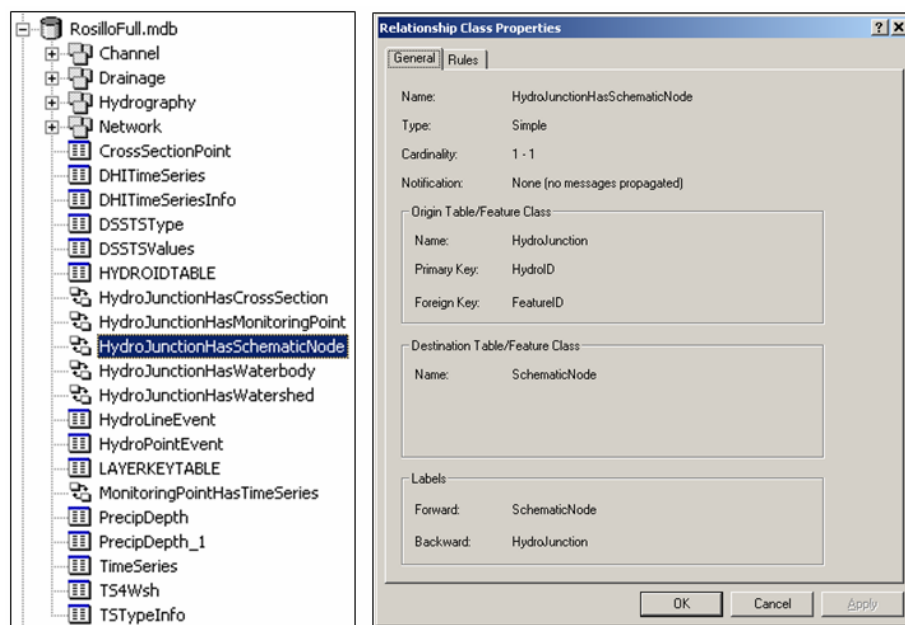


Figure 4.15 Relationship Class for Arc Hydro and HEC-HMS connection

By employing the above HydroJunctionHasSchematicNode relationship, Arc Hydro has access to the HEC-HMS representation inside the GIS, making sure the network topology of HMS is known and honored by the geographic

component and for integration purposes. Besides having access to all the hydrologic elements represented in HEC-HMS, Arc Hydro also needs to select only some of the hydrologic elements. In particular Arc Hydro needs to clearly recognize those HydroJunctions associated with Flow Change Location Points. To accomplish this, the Schematic Nodes representing flow change points (reach entry points of external source hydrographs) have been attributed with the ModelCode (HMSCode in this case) and populated with the corresponding name in HEC-HMS (see Figure 4.16). By doing so, the application, DSS2GDB, can query the relationship “HydroJunctionHasSchematicNode” and select those Nodes inputting a runoff hydrograph for the reach. In short, the hydrologic nodes have been stored in the Arc Hydro SchemaNode feature class, and related to HydroJunctions at the appropriate locations on the stream network based on a relationship class that extends Arc Hydro’s content. Only the HydroJunction features corresponding to flow change points have been labeled with HMSCodes to recognize which are exchange points and which are not.

Shape*	OID	SrcType	HydroID	FeatureID*	HMSCode	FType
Point	15	2	15	441	RC1 RU	Flow Change Location
Point	29	2	29	423	09R CO	Flow Change Location
Point	4	2	4	422	08R CO	Flow Change Location
Point	9	2	9	421	07R CO	Flow Change Location
Point	30	2	30	420	06R CO	Flow Change Location
Point	21	2	21	95	05R CO	Flow Change Location
Point	32	2	32	418	04R CO	Flow Change Location
Point	12	2	12	417	03R CO	Flow Change Location
Point	19	2	19	98	02R CO	Flow Change Location
Point	17	2	17	415	01R CO	Flow Change Location

Figure 4.16 ModelCodes Populated under the SchematicNode Topology

Once the identification of flow change points is achieved, the DSS2GDB application can access the DSS file, search for its corresponding time series records, and transfer them over the geodatabase. Besides being identified by the HMSCode attribute field, the Schematic Nodes have to be associated with its corresponding time series records (runoff hydrographs) in the geodatabase. To provide for the time series association, the DSSTSType table storing the HEC-DSS descriptors for each flow change location is also attributed with a FeatureID field (in the same way as the TimeSeries table) populated with the HydroID of the associated feature having ownership on the data.

FeatureID	DSSFile	A_Part	B_Part	C_Part	D_Part	E_Part	F_Part	sUnit	sType	sTime
35	C:\Vmsproj\Salado_Unified\Salado_Unified.dss	05R CO	FLOW	01 JUL 2002	15MIN	FROM NEXRAD	CFS	INST-VAL	Regular-Interval Time Series Data	0400
98	C:\Vmsproj\Salado_Unified\Salado_Unified.dss	02R CO	FLOW	01 JUL 2002	15MIN	FROM NEXRAD	CFS	INST-VAL	Regular-Interval Time Series Data	0400
415	C:\Vmsproj\Salado_Unified\Salado_Unified.dss	01R CO	FLOW	01 JUL 2002	15MIN	FROM NEXRAD	CFS	INST-VAL	Regular-Interval Time Series Data	0400
417	C:\Vmsproj\Salado_Unified\Salado_Unified.dss	03R CO	FLOW	01 JUL 2002	15MIN	FROM NEXRAD	CFS	INST-VAL	Regular-Interval Time Series Data	0400
418	C:\Vmsproj\Salado_Unified\Salado_Unified.dss	04R CO	FLOW	01 JUL 2002	15MIN	FROM NEXRAD	CFS	INST-VAL	Regular-Interval Time Series Data	0400
420	C:\Vmsproj\Salado_Unified\Salado_Unified.dss	06R CO	FLOW	01 JUL 2002	15MIN	FROM NEXRAD	CFS	INST-VAL	Regular-Interval Time Series Data	0400
421	C:\Vmsproj\Salado_Unified\Salado_Unified.dss	07R CO	FLOW	01 JUL 2002	15MIN	FROM NEXRAD	CFS	INST-VAL	Regular-Interval Time Series Data	0400
422	C:\Vmsproj\Salado_Unified\Salado_Unified.dss	08R CO	FLOW	01 JUL 2002	15MIN	FROM NEXRAD	CFS	INST-VAL	Regular-Interval Time Series Data	0400
423	C:\Vmsproj\Salado_Unified\Salado_Unified.dss	09R CO	FLOW	01 JUL 2002	15MIN	FROM NEXRAD	CFS	INST-VAL	Regular-Interval Time Series Data	0400
441	C:\Vmsproj\Salado_Unified\Salado_Unified.dss	RC1 RU	FLOW	01 JUL 2002	15MIN	FROM NEXRAD	CFS	INST-VAL	Regular-Interval Time Series Data	0400

Figure 4.17 DSS Time Series Catalog in Arc Hydro

The DSS2GDB script tool makes use of the HEC-DSS catalog (.dsc file extension) to obtain a list of all pathnames in the HEC-DSS file and populates existing TimeSeries, TSType, and DSSTSType tables in the extended Arc Hydro geodatabase. The TimeSeries table contains the time stamp and the time series value for a unique combination of FeatureID (site location of dataset) and TSTypeID (variable type).

OBJECTID*	FeatureID*	TSTypeID*	TSDateTime	TSValue
544372	95	2	7/1/2002 1:30:00 PM	990.576233
544373	95	2	7/1/2002 1:45:00 PM	1111.554199
544374	95	2	7/1/2002 2:00:00 PM	1262.369995
544375	95	2	7/1/2002 2:15:00 PM	1420.775635
544376	95	2	7/1/2002 2:30:00 PM	1563.567627
544377	95	2	7/1/2002 2:45:00 PM	1681.781860
544378	95	2	7/1/2002 3:00:00 PM	1776.551514
544379	95	2	7/1/2002 3:15:00 PM	1853.680786
544380	95	2	7/1/2002 3:30:00 PM	1920.758423
544381	95	2	7/1/2002 3:45:00 PM	1982.571533
544382	95	2	7/1/2002 4:00:00 PM	2043.212646
544383	95	2	7/1/2002 4:15:00 PM	2092.558105

Figure 4.18 The Arc Hydro TimeSeries Table

The TSType table describes each variable type stored in the TimeSeries table regardless of the number of features having records for that variable type. Figure 4.19 illustrates an example for generated instantaneous streamflow in CFS. The relevant information for populating this table has to be developed from the HEC-DSS content (pathname parts and headings). It implies that equivalent description values have to be created between the HEC-DSS system and the Arc Hydro time series naming convention.

OBJECTID*	TSTypeID	Variable	Units	IsRegular	TSInterval	DataType	Origin
8278	2	Streamflow	ft^3/s	True	15Minute	Instantaneous	Generated

Figure 4.19 The Arc Hydro TSType Table

As with the GDB2HMSDSS, the first thing done by the application is to open the HEC-DSS file containing the streamflow records for the new HEC-RAS input configuration. As mentioned before, the ZOPEN DSS subroutine is invoked for attaching the source DSS file to the DSS2GDB application. The DSS2GDB application obtains basic information (HEC-DSS descriptors) from the file, allowing reading operations to proceed.

Once the DSS2GDB script tool gets hold of the DSS file, it generates the DSS catalog (list of all pathnames in a DSS file) and stores the pathname list in a catalog file (.dsc file extension) with the same DSS file's name. To generate the catalog the DSS2GDB application calls two DSS subroutines, the ZOPNCA and ZCAT. The ZOPNCA DSS subroutine opens an existing a DSS file's catalog file or creates and opens a new DSS file's catalog file. Once the catalog files have been externally opened, the ZCAT subroutine generates a listing of the record pathnames inside the DSS file.

The DSS2GDB application now has a hold of the DSS file containing the runoff hydrographs but does not know what pathnames/records it must extract from the DSS catalog. Thus, the application needs to establish some type of searching criteria to select the pathnames from the entire catalog that correspond to external source hydrographs at reach entry points. So we have the catalog but do not know what items of it are the ones needed for extraction.

The first filter criterion that the program can establish is for the site location of records/pathnames. The application uses the aforementioned HydroJunctionHasSchematicNode relationship to navigate through all the

SchematicNodes and identify the ones with a valid (not Null) Model Code and store them in a collection (a container of multiple data types) of flow change location points. The collection of flow change points is used to filter the B part in the HEC-DSS catalog. Knowing the B part alone is not enough to classify needed records. To further identify the pathnames associated with hydrographs at flow change points, the application also filters the content of the catalog by the beginning date (D part), the time interval of the time series records (E part), and the optional project descriptor (A Part). This way, the application loops through the entire DSS catalog and selects the time series blocks that meet the filter criteria.

Every time the application navigates the DSS catalog and finds a match for the filter criteria, the application extracts the remaining pathname parts and the headings of the pathname. That is, the application retrieves the variable type (C Part), the user-defined description of the data source (F Part), and the DSS data type (regular or irregular time series, paired/curve data).

When the application finds a pathname that matches the filter criteria for any of the flow change points, it searches the HydroJunction feature class for its corresponding HydroID value (the unique identifier inside the geographic component) and writes that value as the FeatureID in the TimeSeries table. In this way the application now knows the time series value and the feature to which it belongs.

4.1.7 Time Series Extraction

Once the application has defined the data type for a given pathname that satisfies the criteria, the application extracts the time series values based on the appropriate subroutine for the data type found. The possible data types in HEC-DSS and corresponding DSS subroutine for extraction are:

- ZRRTS: DSS subroutine for Reading Regular-Interval Time Series
- ZRITS: DSS subroutine for Reading Irregular-Interval Time Series
- ZRPD: DSS subroutine for Reading Paired Data

The current version of the DSS2GDB script tool implements only the ZRRTS DSS subroutine since the runoff hydrographs generated by HEC-HMS represent regular time series. When a pathname matches the filtering criteria, the application uses the ZRRTS to capture the array of streamflow values for the current flow change point. Since DSS only provides the beginning date and time of the series, the DSS2GDB application loops through every single value in the pathname and computes the corresponding time stamp for it based on the beginning date and time as the starting point and using the time interval to increment the time stamp with respect to the previous value.

Care and consideration must be exercised for the formatting differences between HEC-DSS and the Windows formats of the date and time variable. The application needs to make sure the format conversion between the two systems is appropriate and consistent.

The Windows date format used in an Access table with date fields (and therefore in ArcGIS geodatabases) stores date information under a generic format

based on the short date and time section in the computer's Window's international settings. The time section is used for time intervals less than a day is a 12-hour clock format that uses uppercase A.M. and P.M. This format includes the month, day, year, and for intervals less than 1 day, the hour, minutes, and seconds with its corresponding AM or PM designation. This format and some examples follows: MM/DD/YYYY HH:MM:SS A.M.

Time stamp examples for 15 minute interval time series records will be: 7/1/2002 11:30:00 PM, 7/1/2002 11:45:00 PM, **7/2/2002**, 7/2/2002 12:15:00 AM, 7/2/2002 12:30:00 AM, etc.

For compatibility purposes, it is important to emphasize that for intervals less than a day, the time stamp corresponding to midnight does not show the hour section. It means that the midnight time stamp from the above example, 7/2/2002 12:00:00 A.M. will be displayed in Access as 7/2/2002 as shown above ("the midnight abbreviation").

The HEC-DSS system accepts several date formats for input. Some of those valid formats are: 1 July 2002, July 1, 2002, 01Jul02, 01Jul2002, and 01 Jul 2002. Out of the previous 5 format options, the DDMMYYYY is the most common format one may find labeling HEC-DSS records. The time format in HEC-DSS is specified as a four-digit number representing the 24-hour clock time with an optional colon separating the hours and minutes. Valid entries for time are: 1200, 12:00, 2300, and 23:00.

This difference in date and time format implies that conversion is needed before the transfer is carried over the geodatabase. Specific subroutines to

accomplish this have been developed and incorporated in the data exchange script tools (DSS2GDB, GDB2HMSDSS, and GDB2RASDSS). For this phase of the process, the subroutine converts from HEC-DSS format to Windows date and time format before the values are stored in the geodatabase.

The DSS2GDB application has to actually generate the HEC-DSS time stamps before they are converted to Windows format. For regular time series, the HEC-DSS system only stores the first date and time and then uses the time interval to make sequential time increments based on the interval to generate the other time stamps for the consecutive values. Once the time stamps are generated they can be transferred to the geodatabase using the Windows format.

There is a subtle but quite fundamental difference between the Windows and the HEC-DSS date formats. The comparative Table 4.1 shows an example for the two formats and highlights the compatibility problem.

HEC-DSS Format	Windows Format
01Jul2002 2330	7/1/2002 23:30
01Jul2002 2345	7/1/2002 23:45
01Jul2002 2400	7/2/2002
02Jul2002 0015	7/2/2002 0:15
02Jul2002 0030	7/2/2002 0:30

Table 4.1: Date and Time Formats in Windows and HEC-DSS

The difference in date notation for the above midnight time stamp configures the “midnight duality”. This duality implies two subtle but critical ways of labeling the same point in time. For HEC-DSS the midnight point is

represented by the 2400 hour of the previous day and for Windows it is represented by the 12:00:00 A.M. time stamp of the next day without showing the time stamp (the midnight abbreviation). The aforementioned midnight duality between HEC-DSS and the Windows date notation is a very critical point in terms of data conversion between the two systems. In particular, the programmatic aspects of the midnight duality become quite crucial to ensure the correct transfer of time series records between HEC-DSS and GIS. Figure 4.20 illustrates the schematic of the midnight duality.

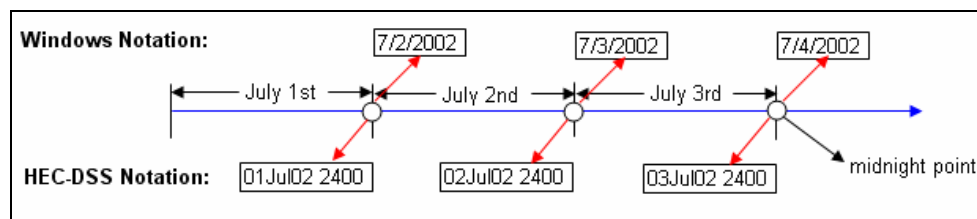


Figure 4.20 The Midnight Duality

Figure 4.20 shows the Windows date notation above the time line for three midnight points at the beginning of July. The Windows notation for midnight corresponds to the 12:00:00 A.M. time stamp of the next day using the midnight abbreviation. The HEC-DSS date notation is shown below the time line. For HEC-DSS the midnight point is label as the 2400 hours of the ending day. Care and consideration for this so called midnight duality was fundamental to achieve a successful framework for transferring time series data and to achieve the intended integration. In other words, format conversion between the two systems was

straight forward for all time stamp points except for the midnight points in time that required special considerations during conversion.

The application extracts not only the time series values itself but also the DSS pathname parts (described in section 4.1.3) and headings (time series units, data type, and the time series type). Based on these HEC-DSS descriptors, the application populates the TimeSeries, TSType, and DSSTSCatalog tables. To accomplish this, equivalent values have to be associated for the application to know what time series descriptor in Arc Hydro corresponds to a given descriptor in HEC-DSS. The following section describes the association between the Arc Hydro and the HEC-DSS descriptors for the purpose of matching and populating the right equivalent and associated values.

4.1.8 HEC-DSS and Arc Hydro Associations

HEC-DSS configures the HEC data storage system for time series, while the Arc Hydro data model for water resources uses tables and relationship classes between them and owning features to define the time series component.

For the DSS2GDB application to be able to transfer information from HEC-DSS to the GIS geodatabase, it needs to be able to translate from DSS nomenclature to Arc Hydro nomenclature. The time series values themselves do not pose a need for conversion since the values are preserved. The time stamp as mentioned before needs to be generated for each value and then converted into Windows date and time format.

The time series descriptors are the most important elements that undergo conversion. The following section provides the details to be considered in this

association which is in turn used for format conversion between the two time series components.

The C Part

The DSS pathname's B part representing the variable type description (e.g., FLOW, PRECIP, STAGE, etc.) needs to be associated with corresponding nomenclature in Arc Hydro. Even though neither HEC-DSS nor Arc Hydro (at least the current version) has adopted a notation for variable type descriptors, the application needs to know how potential notation in DSS will be redefined for the Arc Hydro time series tables. Both HEC-DSS (USACE, 1995) and Arc Hydro (Maidment, 2002) provide some recommendations as to what nomenclature might be used for different types of variables. The following table provides a recommended list for some of the most common variable types used in water resources.

HEC-DSS Notation	Description
PRECIP	Precipitation
PRECIP-INC	Incremental Precipitation for each time interval
PRECIP-CUM	Cumulative Precipitation at end of each time interval
FLOW	Total streamflow discharge
FLOW-ROUT	Routed flow
FLOW-DIVERT	Flow diverted out of the river at this location
FLOW-IN	Reservoir inflow
FLOW-OUT	Reservoir outflow
ELEV	Relative water surface elevation
ELEV-RES	Reservoir relative elevation
STORAGE	Reservoir storage
STAGE	Absolute water surface elevation
EVAPORATION	Reservoir evaporation
EVAP-PAN	Pan evaporation

Table 4.2: HEC-DSS Recommended Variable Type Notation

As opposed to HEC-DSS, the Arc Hydro definition for variables type is quite flexible. It is up to each user and project at hand that the variable types are established with basically no restrictions on the notation itself.

Arc Hydro Notation
NEXRAD Rainfall
Design Rainfall
Streamflow
Gage height
Storage
Elevation

Table 4.3: Arc Hydro Recommended Data Parameter Notation

The HEC-DSS system of the other hand, recommends the use of a variable type notation that is as abbreviated and unique as possible capable of defining the subtype of variable type. For instance, the PRECIP-CUM variable type notation not only defines the variable type as precipitation but also that the values correspond to the total depth fallen at the end of each time interval. In turn, the Arc Hydro system allows for a non-abbreviated notation (Variable field equal to Rainfall in this case) and leaves the subtype definitions for the other descriptors (the TSDataType field descriptor equal to Cumulative in this case).

These differences in approach come from the philosophy and structure of the systems. HEC-DSS storage and retrieval efficiency relies on the unique composition of the pathname parts to rapidly identify the needed record. To accomplish this, distinguishing naming convention at all levels (pathname parts) promotes a faster retrieval of data records by empowering a filtering search

through fewer records. That is, if specific flow subtypes are defined, the search for a given subtype of flow will be among fewer records. Also, given that the maximum length of a DSS pathname is 80 characters, with each part not exceeding 32 characters, each part definition must be as abbreviated as possible. In short, the HEC-DSS descriptors must promote uniqueness (even at the subtype level) and be abbreviated to comply with pathname limitations.

The Arc Hydro definition for time series, in particular its descriptors in the TSType object class have no limitation in length, and the uniqueness is achieved by the combination of all descriptors which gets synthesized by a unique TSTypeID.

For the integration in Map2Map, the variable types that need to be associated are the radar precipitation and the streamflow. The corresponding equivalences used for achieving the integration are:

HEC-DSS	Arc Hydro
PRECIP	Rainfall
PRECIP-INC	Rainfall
PRECIP-CUM	Rainfall
FLOW	Streamflow
FLOW-ROUT	Streamflow
FLOW-DIVERT	Diversion
FLOW-IN	Inflow
FLOW-OUT	Outflow
ELEV	Elevation
ELEV-RES	Elevation
STORAGE	Storage
STAGE	Stage
EVAPORATION	Evaporation
EVAP-PAN	Evaporation

Table 4.4: Variable Type Equivalence between HEC-DSS and Arc Hydro

The B Part

The DSS pathname's B part represents the name of the element that owns the data (data's site location). The name of the modeling element (in HEC-HMS in this case) is normally assigned to the pathname's B Part. In Arc Hydro, the TimeSeries table storing the values and associated time stamp are related to a geographic feature by means of a relationship class that is defined by populating the FeatureID field in the TimeSeries table with the HydroID of the feature that has ownership over the times series records. In addition to this, the DSSTSCatalog table listing all the HEC-DSS records also contains the FeatureID field with the HydroID of the feature it belongs. On implementation, the DSS2GDB application extracts the B Parts from the HEC-DSS file that match the flow change points in HEC-HMS (as stored in the SchematicNode feature class, see section 3.4.3) gets the HydroIDs from the related HydroJunctions, and stores the corresponding HydroID of the HydroJunction (not the Schematic Node) in the FeatureID field of the DSSTSType table. Thus, the B Part is searched based on the SchematicNode definition of flow change points and then used to populate the B part field of the DSSTSType table.

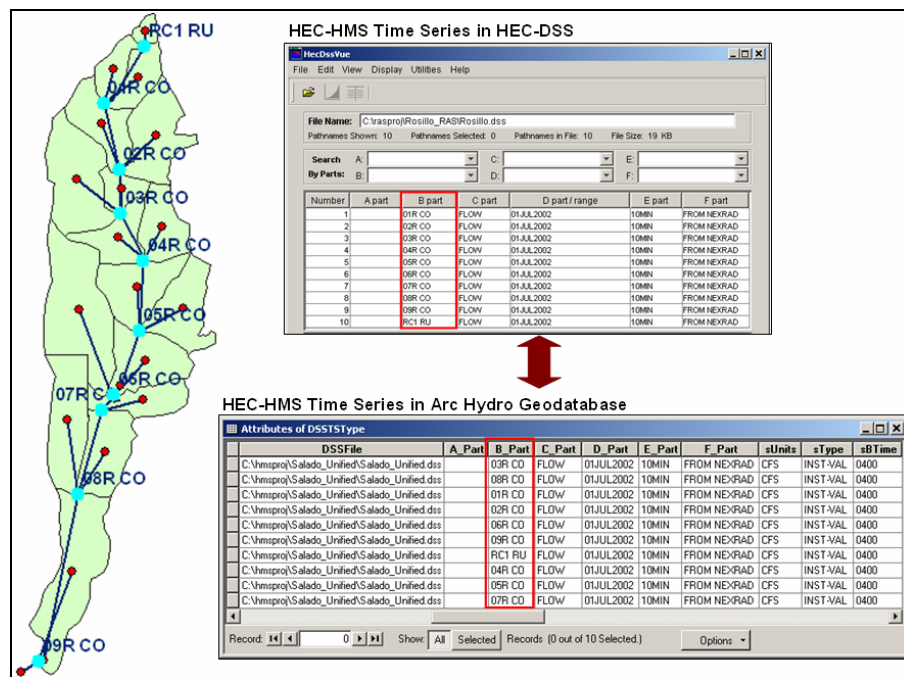


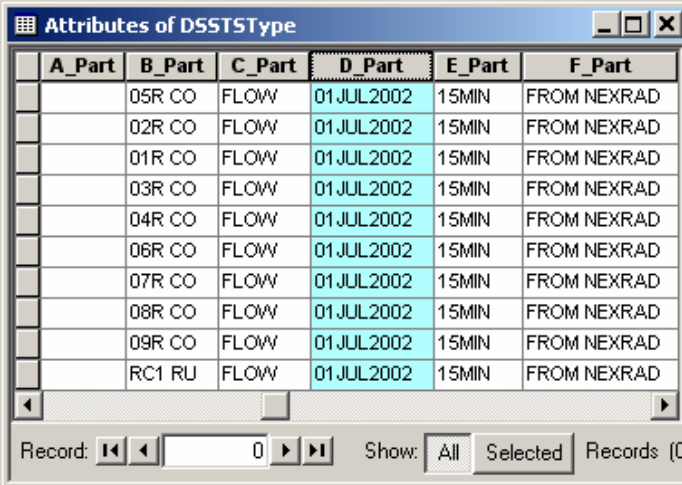
Figure 4.21 Filtering HEC-DSS records through Arc Hydro

The Figure 4.21 depicts how the geographic framework is used to filter and select the required HEC-DSS entries and transfer the B part site location to the DSSTSType table in Arc Hydro to fully reference the DSS records. Thus, once the B part is selected from the geographic framework, the DSS pathnames are selected and the extracted B parts used to populate DSSTSType in Arc Hydro.

The D Part

The starting date of the data block (HEC-DSS card) is also extracted from the DSS catalog. Once the DSS records representing flow change points are identified (B part filter), by using the geographic framework, the D part is extracted from the DSS catalog file (.dsc file generated by the ZOPNCA and

ZCAT DSS subroutines) and stored in the DSSTSType catalog table in Arc Hydro.



A_Part	B_Part	C_Part	D_Part	E_Part	F_Part
	05R CO	FLOW	01JUL2002	15MIN	FROM NEXRAD
	02R CO	FLOW	01JUL2002	15MIN	FROM NEXRAD
	01R CO	FLOW	01JUL2002	15MIN	FROM NEXRAD
	03R CO	FLOW	01JUL2002	15MIN	FROM NEXRAD
	04R CO	FLOW	01JUL2002	15MIN	FROM NEXRAD
	06R CO	FLOW	01JUL2002	15MIN	FROM NEXRAD
	07R CO	FLOW	01JUL2002	15MIN	FROM NEXRAD
	08R CO	FLOW	01JUL2002	15MIN	FROM NEXRAD
	09R CO	FLOW	01JUL2002	15MIN	FROM NEXRAD
	RC1 RU	FLOW	01JUL2002	15MIN	FROM NEXRAD

Record: 0 Show: All Selected Records (C

Figure 4.22 D part in Arc Hydro's DSSCatalog

The E Part

The time interval of the stored time series data is also extracted from the DSS catalog file (.dsc file extension) and stored in the DSSTSType catalog table in Arc Hydro.

A_Part	B_Part	C_Part	D_Part	E_Part	F_Part
	05R CO	FLOW	01.JUL2002	15MIN	FROM NEXRAD
	02R CO	FLOW	01.JUL2002	15MIN	FROM NEXRAD
	01R CO	FLOW	01.JUL2002	15MIN	FROM NEXRAD
	03R CO	FLOW	01.JUL2002	15MIN	FROM NEXRAD
	04R CO	FLOW	01.JUL2002	15MIN	FROM NEXRAD
	06R CO	FLOW	01.JUL2002	15MIN	FROM NEXRAD
	07R CO	FLOW	01.JUL2002	15MIN	FROM NEXRAD
	08R CO	FLOW	01.JUL2002	15MIN	FROM NEXRAD
	09R CO	FLOW	01.JUL2002	15MIN	FROM NEXRAD
	RC1 RU	FLOW	01.JUL2002	15MIN	FROM NEXRAD

Record: 0 Show: All Selected Records (C

Figure 4.23 E part in Arc Hydro's DSSCatalog

The equivalence for the time interval (E part) notation between Arc Hydro and HEC-DSS is shown in Table 4.5. There is basically a match between the time interval definitions of the two systems since the Arc Hydro time intervals were taken from HEC-DSS.

HEC-DSS	Arc Hydro
1MIN	1Minute (Code = 1)
2MIN	2Minute (Code = 2)
3MIN	3Minute (Code = 3)
4MIN	4Minute (Code = 4)
5MIN	5Minute (Code = 5)
10MIN	10Minute (Code = 6)
15MIN	15Minute (Code = 7)
20MIN	20Minute (Code = 8)
30MIN	30Minute (Code = 9)
1HOUR	1Hour (Code = 10)
2HOUR	2Hour (Code = 11)
3HOUR	3Hour (Code = 12)
4HOUR	4Hour (Code = 13)
6HOUR	6Hour (Code = 14)
8HOUR	8Hour (Code = 15)
12HOUR	12Hour (Code = 16)
1DAY	1Day (Code = 17)
1WEEK	1Week (Code = 18)
1MON	1Month (Code = 19)
1YEAR	1Year (Code = 20)
	Other (Code = 99)

Table 4.5: Time Interval Equivalence between HEC-DSS and Arc Hydro

The F Part

The F Part that usually represents a unique descriptor of the data's origin (either the obtaining method or the archiving agency) is also extracted from the DSS catalog file (.dsc file extension) and stored in the DSSTSType catalog table in Arc Hydro.

A_Part	B_Part	C_Part	D_Part	E_Part	F_Part
	05R CO	FLOW	01 JUL 2002	15MIN	FROM NEXRAD
	02R CO	FLOW	01 JUL 2002	15MIN	FROM NEXRAD
	01R CO	FLOW	01 JUL 2002	15MIN	FROM NEXRAD
	03R CO	FLOW	01 JUL 2002	15MIN	FROM NEXRAD
	04R CO	FLOW	01 JUL 2002	15MIN	FROM NEXRAD
	06R CO	FLOW	01 JUL 2002	15MIN	FROM NEXRAD
	07R CO	FLOW	01 JUL 2002	15MIN	FROM NEXRAD
	08R CO	FLOW	01 JUL 2002	15MIN	FROM NEXRAD
	09R CO	FLOW	01 JUL 2002	15MIN	FROM NEXRAD
	RC1 RU	FLOW	01 JUL 2002	15MIN	FROM NEXRAD

Figure 4.24 F part in Arc Hydro's DSSCatalog

The Units

The units of the time series values are an output of the ZRRTS DSS subroutine that is also stored as a field value for each of the flow change location records in the DSSTSType table.

B_Part	C_Part	D_Part	E_Part	F_Part	sUnits
05R CO	FLOW	01 JUL 2002	15MIN	FROM NEXRAD	CFS
02R CO	FLOW	01 JUL 2002	15MIN	FROM NEXRAD	CFS
01R CO	FLOW	01 JUL 2002	15MIN	FROM NEXRAD	CFS
03R CO	FLOW	01 JUL 2002	15MIN	FROM NEXRAD	CFS
04R CO	FLOW	01 JUL 2002	15MIN	FROM NEXRAD	CFS
06R CO	FLOW	01 JUL 2002	15MIN	FROM NEXRAD	CFS
07R CO	FLOW	01 JUL 2002	15MIN	FROM NEXRAD	CFS
08R CO	FLOW	01 JUL 2002	15MIN	FROM NEXRAD	CFS
09R CO	FLOW	01 JUL 2002	15MIN	FROM NEXRAD	CFS
RC1 RU	FLOW	01 JUL 2002	15MIN	FROM NEXRAD	CFS

Figure 4.25 Unit Definition in Arc Hydro's DSSCatalog

There are not specific valid values for the value units of the time series records in HEC-DSS or Arc Hydro. In practice, however, some notation is more standard in engineering projects. Some of these standard notations are presented in Table 4.6.

HEC-DSS	Arc Hydro	Description
CMS	cms	cubic meters per second
CFS	cfs	cubic feet per second
MM	mm	millimeters
INCHES	inch	inches
METERS	meters	meters
FEET	ft	feet

Table 4.6: Unit Notation Equivalence between HEC-DSS and Arc Hydro

The Data Type

The Arc Hydro time series data type defines several categories of the time series records. For instance, if the time series value corresponds to an instant in time the values are classified as instantaneous. If the values represent the accumulated measurement since the beginning of the event being monitored the values are classified as cumulative. If the values represent the amount recorded for each time interval only without regard for the starting time in the past the records are classified as incremental. If the values represent the average rate over a time interval the time series records are classified as average. This criterion for classifying time series records is present in all time series systems including Arc Hydro and HEC-DSS under different notations. HEC-DSS stores the data type as a header of the pathname card and not as a pathname part. It is extracted for each

pathname card using the ZRRTS DSS subroutine which returns the data type as an output argument. Arc Hydro stores the data type descriptor in the TSType table under the DataType field name and using a coded value domain.

The corresponding equivalences between the data type classifications in Arc Hydro and HEC-DSS are shown in Table 4.7.

HEC-DSS	Arc Hydro
INST-VAL	Instantaneous (code = 1)
INST-CUM	Cumulative (code = 2)
PER-CUM	Incremental (code = 3)
PER-AVER	Average (code = 4)
Not defined	Maximum (code = 5)
Not defined	Minimum (code = 6)

Table 4.7: Data Type Equivalence between HEC-DSS and Arc Hydro

The DSS2GDB application uses the above system equivalences to transfer DSS records over to the Arc Hydro geodatabase by properly populating the DSSTSType and TSType table.

The Time Series Type

Time series records can be measured or observed at regularly or irregularly spaced intervals. This classifying criterion is supported by both HEC-DSS and Arc Hydro. The HEC-DSS system stores the data type as a header of the pathname card and not as a pathname part. After the DSS file has been open and the DSS catalog has been created, the DSS2GDB application uses the ZDTYPE DSS subroutine to obtain the time series type for any of the pathnames inside the

catalog list. For the DSS2GDB application, the program selects those pathnames associated with flow change locations and extracts the time series type using the ZDTYPE DSS subroutine to decide what subroutine should be used for extracting the time series values and to define the time series type descriptor in Arc Hydro. Even though the time series used and generated for the Map2Map application are regularly spaced intervals, the application checks for other types of time series to provide a generic approach.

In Arc Hydro the time series type it is stored in the TSType table under the IsRegular Boolean field. The corresponding equivalences between the time series type classifications in Arc Hydro and HEC-DSS are shown in Table 4.8.

HEC-DSS	Arc Hydro
Regular-Interval Time Series Data	IsRegular Boolean = 1 (True)
Irregular-Interval Time Series Data	IsRegular Boolean = 2 (False)

Table 4.8: Time Series Type Equivalence between HEC-DSS and Arc Hydro

After the ZRRTS DSS subroutine is executed, and the time series are extracted from HEC-DSS, the application invokes ZCLOSE, the DSS software subroutine in charge of releasing control of a DSS file from an external program. This closing subroutine is quite important to make sure the DSS file becomes available to other external programs. From the programmatic perspective this subroutine must be carefully implemented since it should be invoked even if an error is generated during execution. That is, the error handling component must

implement a call to the ZCLOSE subroutine to properly release control over the source DSS file.

4.1.9 The GDB to RASDSS Script Tool

The ability to read data from HEC-DSS has been also added to HEC-RAS in order to extract flow and stage data for use in water surface profile calculations. The extraction of flow data is normally done from DSS files generated by HEC-HMS simulations over the same watershed study area under hydraulic analysis.

The GDB2RASDSS script tool transfers time series data from the geodatabase to a HEC-RAS DSS binary file. Additionally, the tool updates the HEC-RAS project file (.prj), plan file (.p0*), and flow file (.f0*) to reflect the new time series records being fed by the geographic integration. This tool is run prior to HEC-RAS execution, to provide streamflows at certain cross sections representing flow exchange points.

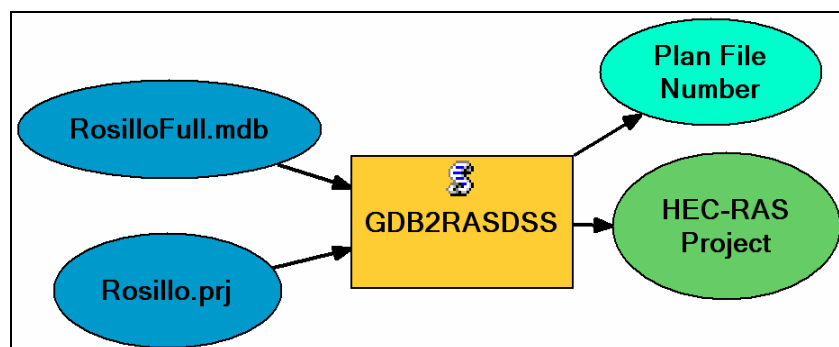


Figure 4.26 GDB2RASDSS Model Component

4.1.9.1 Opening the HEC-DSS file

The first thing done by the GDB2RASDSS script tool is to open or generate the HEC-DSS file that stores the streamflow records for the new HEC-RAS input configuration. The ZOPEN DSS subroutine is used for attaching the new or existing DSS file to the GDB2RASDSS application and also for obtaining basic information from the file that allows the application to write new records to it. As mentioned before, the ZOPEN is the first DSS subroutine to be called when interfacing with the HEC-DSS system. Again, the DSS file does not need to exist before the ZOPEN subroutine is called and it will be created if it is not present.

4.1.9.2 Reading Time Series Information

The next section of the application is in charge of reading the Arc Hydro time series table and the DSSTSType with all the HEC-DSS descriptors for each flow change location. By reading this information the application is able to write into a new or existing HEC-DSS file, and write the new flow file based on the new flow hydrographs and for the HEC-RAS setup.

The application reads the DSSTSType table to get all the DSS descriptors for each pathname containing records for a flow change location from the DSS catalog. The DSS descriptors (Figure 3.21) are stored in a 2-D array in which the first dimension contains each flow change location point and the second dimension contains each of the needed DSS descriptors. The DSS descriptors needed for the writing of records into a HEC-DSS file are described in Table 4.9.

HEC-DSS Descriptor	Description
Target DSS Filename	Directory Location of DSS file
Pathname	Complete DSS Pathname
HMSCode (B part)	Site ID with Ownership on Data
Beginning Date (D Part)	Date Stamp of First Record
Beginning Time	Time Stamp of First Record
Data Units	Measured Units
Data Type (Time Extent)	Measuring Method
Time Interval (E Part)	Data Time Increments
FeatureID	Spatial Feature named by Site ID

Table 4.9: HEC-DSS Descriptors for Data Exchange

The GDB2RASDSS application reads the TimeSeries table in Arc Hydro to get the time series values for each flow change location. The values are also stored in the GDB2RASDSS tool using a 2-D array in which the first dimension represents the number of the flow change point and the second dimension represents the number of the time series value. The GDB2RASDSS application loops through each flow change location in the time series table implementing a query filter definition to ensure efficiency in the search process and that uses the FeatureID (HydroID of the HydroJunction) of the flow change point as the filter criterion.

4.1.9.3 Accessing the CrossSection Feature Class

In the same manner as the Schematic Nodes (for the hydrologic representation in GIS), the Cross Sections describing the geometry of the river

channel (the hydraulic representation) are also related to HydroJunction points on the Arc Hydro stream network.

Through the permanent HydroJunction relationship in Arc Hydro (HydroJunctionHasCrossSection), the next downstream CrossSection from a given SchemaNode can be located. Therefore, each runoff hydrograph can be matched to the cross section closest (downstream) to the hydrologic node where that streamflow addition occurs in the network.

The HydroJunctionHasCrossSection relationship class is used to locate the cross section receiving a hydrograph source and to extract the RASCode of the cross section. The RASCode is defined as the concatenation of the StreamID, ReachID, and Station, e.g., RASCode = Rosillo,Upper,55649.9.

4.1.9.4 Generating the Flow File for HEC-RAS

Entry points of external source runoff hydrographs from tributary subbasins are defined in HEC-RAS by cross sections. For the purpose of an automated scheme, the entering hydrographs must be stored in HEC-DSS and fed into HEC-RAS. An HEC-DSS data connection must be established for all the flow change point locations represented by receiving cross sections. The Flow file is in charge of establishing connections between HEC-RAS cross-section locations and pathnames contained in the HEC-DSS file. By establishing the connection, flow data is extracted for HEC-RAS and from the hydrographs at each of the locations being read from DSS. This implies that the Flow file needs to be re-written at run time to establish the connection between an HEC-RAS cross section and a particular pathname in the DSS file feeding flow series.

In order to define the flow file for HEC-RAS, the GDB2RASDSS application needs to extract the peak streamflow values from each entering hydrograph. The DSS2GDB component is used to get an overall peak flow for each profile computation in HEC-RAS. For unsteady definitions, the peak extraction can be eliminated and the entire hydrograph will be used to define the unsteady flow configuration. However, unsteady definitions in HEC-RAS provide a higher level of complexity since it only allows definition of a single reach at a time. By having a single reach definition, the unsteady state definitions will hamper the ability to define boundary conditions for entering tributaries providing additional runoff hydrographs to be routed along the main channel system. To work around this unsteady restriction several “Map2Map” definitions for each reach can be developed and nested to execute sequentially by means of a script in charge of establishing the Map2Map batch process. Even though viable, the aforementioned alternative to configure unsteady simulations via nested Map2Map models for each reach may prove to be cumbersome and not robust under the current version of ArcGIS ModelBuilder.

For the Map2Map application to run an automated Hydrologic and Hydraulic configuration, the execution of HEC-RAS must be accomplished without having to rely on its graphical user interface. HEC-RAS programs can be externally executed (as described in next section 4.1.10) but the execution is done for the default settings stored inside the model configuration. In other words, if the Map2Map integration is generating new values for an existing HEC-RAS model setup those new values need to be integrated in the default components of

the model if the new configuration is to be executed under an automated mode of operation. Given that the seamless and end-to-end integration proposed by Map2Map requires an automated mode of operation, the new input generated through the integration needs to become part of HEC-RAS default settings.

Next, the GDB2RASDSS application reads the HEC-RAS project file (.prj file extension) corresponding to the HEC-RAS configuration that is to be used for simulation and finds the current or executed by default plan file (.p# file extension). The current plan file contains a unique combination of geometry and flow files that are used when the model is run under default settings. In general, a HEC-RAS project file contains registries for all geometry and flow files representing multiple scenarios for the same river-reach system. In the project file there is also a registry for the current plan file that will be executed under default settings. The plan file is made up of a unique combination of geometry and flow files describing a given river channel setup and a given flow profile going through it. Figure 4.27 illustrates the potential combinations in a HEC-RAS project between various geometry and flow files to define various plan files and how the output files are associated with each of the plan files simulated and stored in the project.

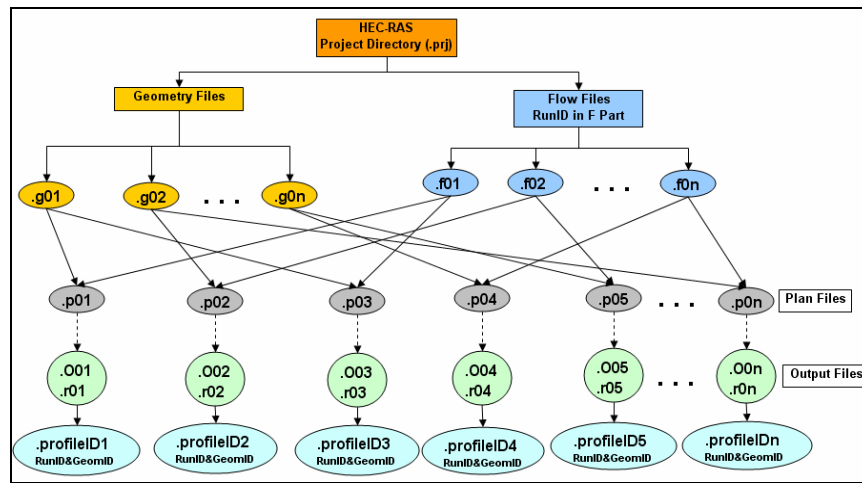


Figure 4.27 Geometry and Flow Combinations for Plan Configurations in RAS

After the current plan file is found in the project file registry, the GDB2RASDSS application reads the current plan file in turn to find the registered flow file. That registered flow file corresponds to the default flow file that will be used during HEC-RAS execution (unless the graphical user interface is used to change the default settings). After the flow file in the current plan file is found, the GDB2RASDSS application rewrites the current flow file based on the new flow information coming from the Map2Map integration scheme. This way, the registered flow file is rewritten with a new definition derived from the precipitation event being transformed and ingested by the Map2Map workflow.

Once the current flow file is updated, the application uses a FORTRAN DSS subroutine that writes regular time series into HEC-DSS, the ZSRTS subroutine (Store Regular-Interval Time Series Data). This subroutine writes a new DSS binary file using the 2-D arrays containing the time series DSS descriptors and the time series values. The subroutine has been incorporated into

the main program as a public module that is invoked every time it is required. The main input and output arguments that are passed through this utility are described below denoting the descriptors needed to store datasets in the HEC-DSS system. This description is of particular relevance for the definition of all the parameters that have to be present in the GIS system before the data exchange operation takes place. The detailed descriptions of the ZSRTS DSS subroutine were provided in section 4.1.3.1.

After the ZSRTS subroutine is executed, the GDB2RASDSS application invokes ZCLOSE, another DSS software subroutine that updates and releases a DSS file from a program. This closing subroutine is quite important to make the DSS file available to other programs like HEC-RAS itself (next process in workflow).

4.1.10 The HEC-RAS Caller Script Tool

The second application that needs to be externally executed is the hydraulic model, HEC-RAS. As opposed to HEC-HMS which does not expose its functionalities to client applications (only execution by command-line statements from a script), HEC-RAS has been packaged in a way that allows several internal tasks to be executed from a client application.

The RASCaller script tool is in charge of externally call HEC-RAS to calculate the water surface elevation and extent at each defined Cross Section. The tool generates a RASexport.sdf SDF file (Spatial Data File for GIS connection) and returns control to ArcGIS.

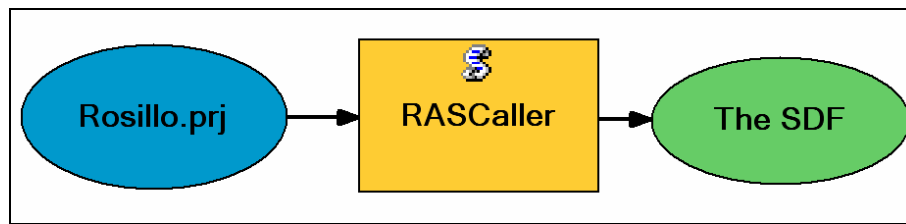


Figure 4.28 RASCaller ModelBuilder Component

The execution of HEC-RAS is done through the HEC-RAS object library that exposes some execution and writing functionalities. The ras.exe application for HEC-RAS version 3.1.1 and above is packaged as an ActiveX EXE component that provides reusable code in the form of objects. This application format for HEC-RAS allows external applications (clients), like the RASCaller process in Map2Map, to instantiate objects and calls their properties and methods. The objects are designed to divide and make available the main functionalities of the component (the HEC-RAS model in this case).

An ActiveX EXE is a special type of COM component (Component Object Model is a protocol that enables software components to communicate to each other) that can be linked to other components to build applications. One of the best known examples of COM Object Linking and Embedding (OLE) technology implementation is the ability of Word documents to dynamically link to data in Excel spreadsheets. An ActiveX EXE is both an EXE file (it is loaded into its own address space and given its own process and threads) and it is also designed to be an OLE automation server (an ActiveX DLL). As with any ActiveX DLL, it exposes interfaces (classes) to be used by a client application (the RASCaller in Map2Map in this case).

Then, the RASCaller application accesses the exposed functionality in HEC-RAS by making a reference to the “HEC River Analysis System” application (HEC-RAS) which becomes available as a COM component after HEC-RAS is installed in the local computer.

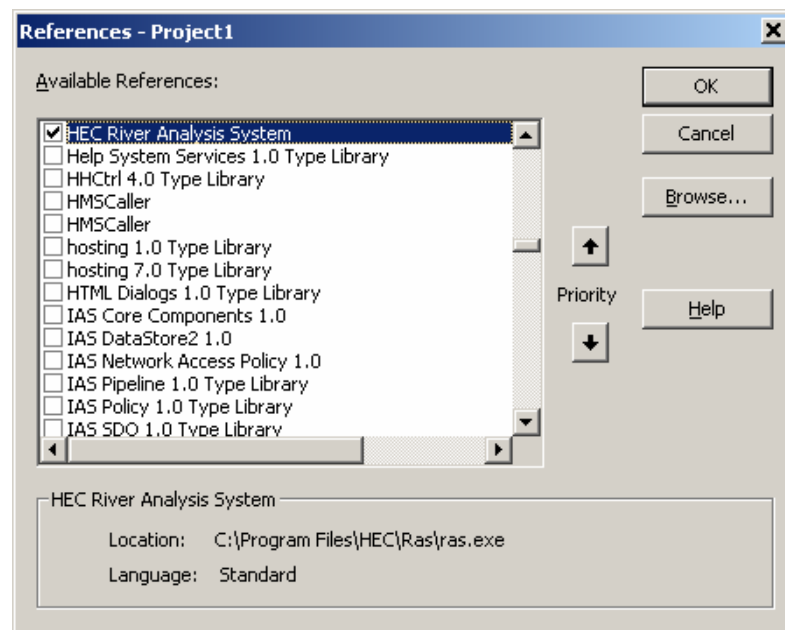


Figure 4.29 HEC-RAS Object Library Reference in Visual Basic

After the application makes the reference to the HEC-RAS library, all the classes (objects) in that library become available to the application (the RASCaller application) and a great deal of additional functionality is gained through the properties, methods, and events of those classes.

The HECRASController class is the main object used in the RASCaller to access HEC-RAS’s exposed functionality. The HECRASController provides 32

methods and 3 properties to the client application. Figure 4.30 provides a list of the HECRASController's class members.

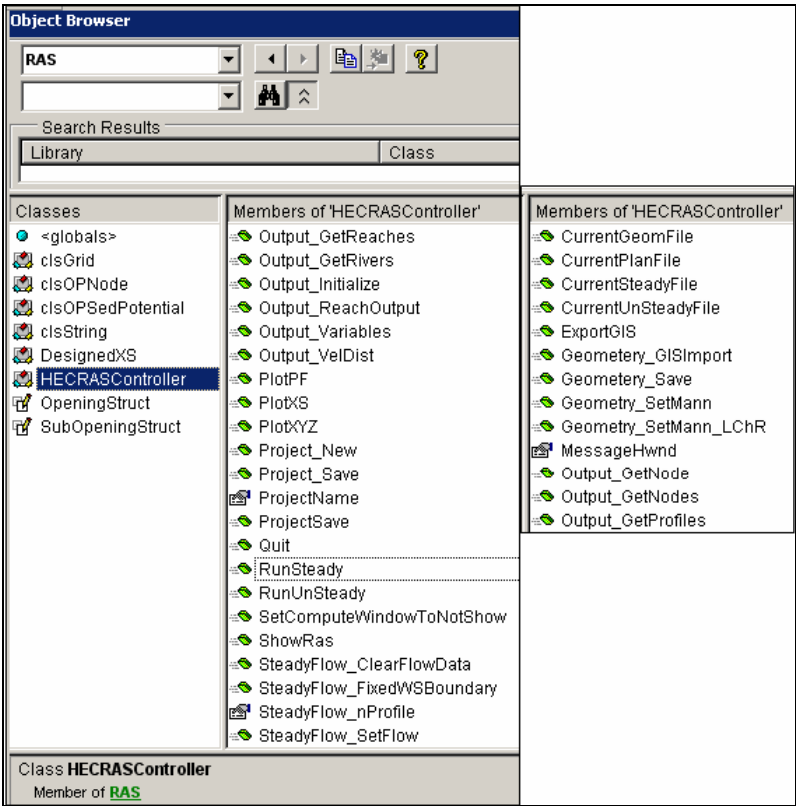


Figure 4.30 HEC-RAS Object Library and Components

The first member used by the RASCaller application is the ProjectName property. Through this property, the RASCaller application assigns the HEC-RAS project name that it wants to be interfaced with. As shown in Figure 4.31 the ProjectName property is a member of the HECRASController's class inside the RAS object library and the project name must be passed as a string.

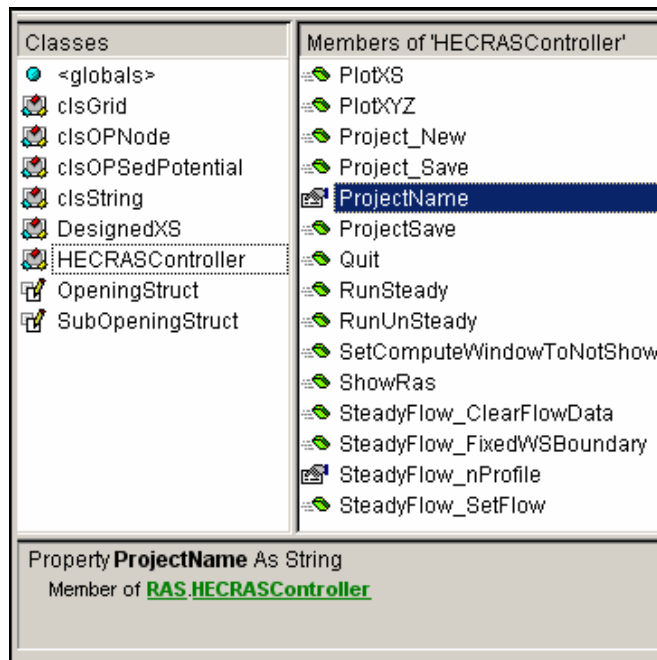


Figure 4.31 ProjectName Property in RAS Library

The second member used by the application is the SetComputeWindowToNotShow method. The use of this method is optional and was implemented for the RASCaller in Map2Map to avoid displaying the HEC-RAS processing window. By eliminating the number of windows open at processing time over ModelBuilder window already open, some delays in screen visualization are avoided by not having to wait for the screen to refresh and continue with the other processes in the workflow. That is, the next process after HEC-RAS is executed will not be “blurred” by the delayed refresh from the previous processing window. Figure 4.32 shows the signature of the subroutine representing the method and as shown it does not require arguments.

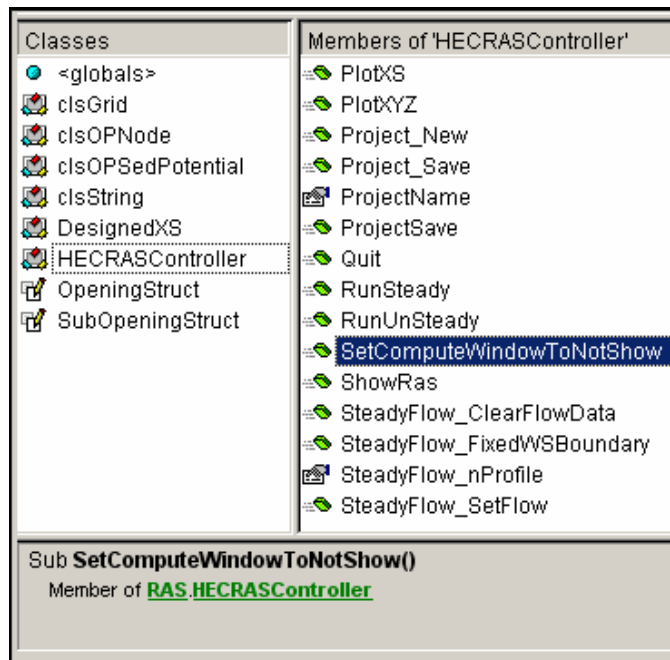


Figure 4.32 SetComputeWindowToNotShow Method in RAS Library

After the HEC-RAS project has been assigned and the option for not opening the processing window has been selected, the application invokes the RunSteady method to execute a steady state flow simulation. By means of this method, HEC-RAS is executed and water surface elevations and spatial extents are obtained for each cross section defining the river channel geometry.

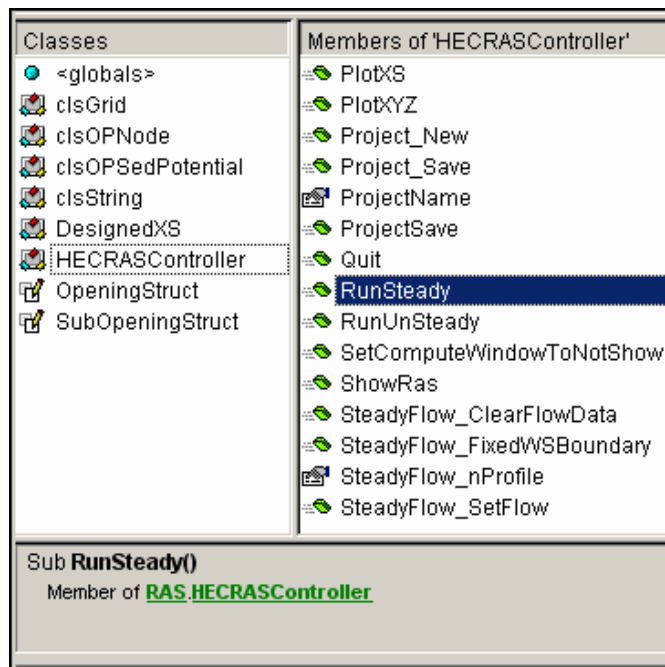


Figure 4.33 RunSteady Method in RAS Library

Once HEC-RAS is executed it generates two output files: one binary file (.Oxx file extension) and one ASCII file (.rxx file extension). The binary file can only be interpreted by the HEC-RAS application and the ASCII file even though it contains readable text format, this is not presented in a self explanatory format that allows for direct interpretation. HEC-RAS, however, contains an option that allows its output to migrate to a geographic information system (GIS). By means of this option, HEC-RAS exports hydraulic information to GIS (water surface elevations, velocities, and so on) having a geometric character. The export to GIS option generates the SDF file, Spatial Data File, which can then be ingested in GIS as described in appendix B of the HEC-RAS User's Manual (USACE, 2002b).

The origin of the SDF file goes back to the first version of the HEC-GeoRAS interface (v.3.1) which was first developed in 1996 as an extension for ArcView 3.2. It provides a set of procedures, tools, and utilities for the preparation of GIS data for import into RAS and generation of GIS data from RAS output (USACE, 2002a).

HEC-RAS writes the export file with the RASexport.sdf file extension (for HEC-GeoRAS to generate GIS data from RAS output) and the information content of the SDF file can be migrated into GIS for further spatial manipulations, analysis, and presentations. For HEC-RAS to be able to generate an SDF file the geometry of the RAS project must be fully georeferenced. Two conditions need to be met for a geometry component to be georeferenced:

- All reaches must have an X-Y vertex definition
- All cutlines (plan view representation of cross sections) must have an X-Y vertex definition

Thus, if the HEC-RAS project is fully georeferenced (normally the case when the RAS input is created from GIS data) its output can be exported into an SDF file for GIS ingestion. The ExportGIS subroutine of the HECRASController is the method invoked by the RASCaller application to generate the RASexport.sdf file containing the hydraulic output information. Figure 4.34 depicts the signature of the ExportGIS method.

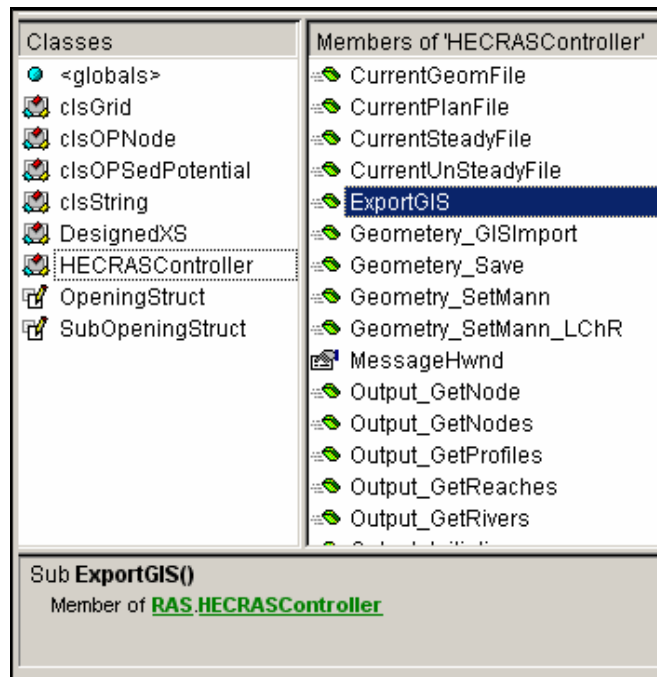


Figure 4.34 ExportGIS Method in RAS Library

Finally, once HEC-RAS has been executed and its output transferred into a “GIS compatible” format, the application uses the Quit method to save the project and exit the HEC-RAS application. The signature of the subroutine is shown in Figure 4.35.

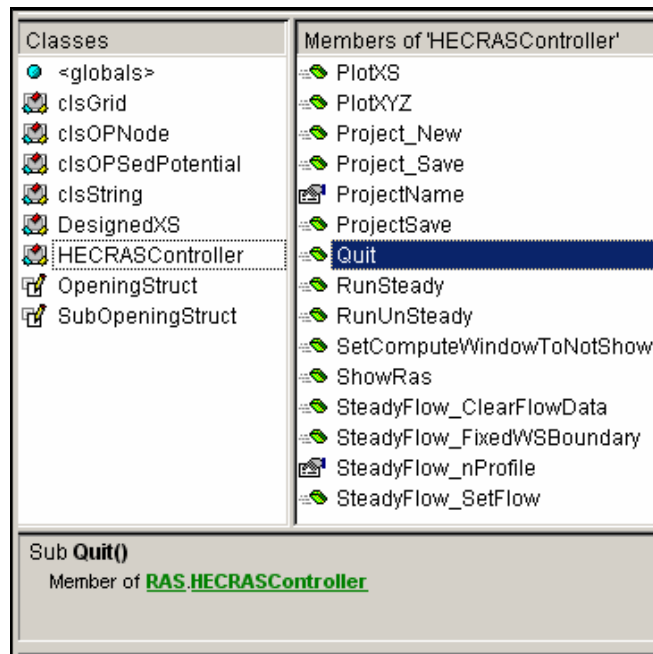


Figure 4.35 Quit Method in RAS Library

4.1.11 The SDF2XML Script Tool

After the hydraulic simulation is completed in HEC-RAS, the water surface elevations and their geometric definition (cross section location and extent) need to be transferred back to GIS for generating the flood inundation map.

As mentioned before, an SDF file (Spatial Data File) is the GIS compatible output file from a HEC-RAS simulation. This file contains geometric definitions of all relevant elements (reaches and cross section cutlines) as well as water surface elevations and spatial extent (flood boundary line) for each cross section in the HEC-RAS model. Exchange of information between external systems has proved to be a challenge due to the great diversity of file formats and

data structures delivered by every system. One alternative that has been adopted as a standard for data exchange between independent applications is XML (Extended Markup Language), a language that temporarily holds data for moving it between heterogeneous applications and platforms by clearly “tagging” its data content. To overcome the distributed and heterogeneous character of the datasets, federated database management systems (DBMS) constituting seamless approaches to data exchange have been proposed (Raman et al., 2002). In general, the data transfer between applications must be as decoupled from the applications as possible while still maintaining context. XML provides the means for accomplishing this (Harold and Means, 2001).

XML is a self-describing mark-up language that describes hierarchical trees of data. The description in XML is done through HTML type of tags and they describe the data that they delimit as opposed to describing the look of the data they delimit. Thus, the describing tags hold the metadata keywords that tell applications the meaning of the data and not the visual representation of it. The tags can contain multiple levels of metadata descriptors in the form of node names and attributes.

Another key characteristic of XML is its extensibility character. This means that unlike HTML, which has a limited number of predefined tags describing the look of the data (bold, italic, table, and so on), XML supports custom tags. So, if the data in XML is related to water resources, the engineer may create specific tags for that type of information. Figure 4.36 shows an

example XML file holding data for the Rosillo Creek in San Antonio having a tag definition customized for water resources.

```
- <CrossSection>
  <StreamID Value="Rosillo" />
  <ReachID Value="Upper" />
  <Station Value="95823.01" />
- <CutLine>
  <CutLinePoint X="2170824.738" Y="13730871.054" />
  <CutLinePoint X="2170803.277" Y="13730875.972" />
  <CutLinePoint X="2170760.576" Y="13730886.032" />
  <CutLinePoint X="2170712.51" Y="13730897.434" />
  <CutLinePoint X="2170657.961" Y="13730910.177" />
  <CutLinePoint X="2170597.376" Y="13730924.485" />
  <CutLinePoint X="2170566.301" Y="13730931.639" />
  <CutLinePoint X="2170509.516" Y="13730945.053" />
  <CutLinePoint X="2170422.55" Y="13730965.397" />
  <CutLinePoint X="2170388.122" Y="13730973.445" />
  <CutLinePoint X="2170339.609" Y="13730984.847" />
  <CutLinePoint X="2170286.177" Y="13730997.59" />
  <CutLinePoint X="2170241.241" Y="13731008.097" />
  <CutLinePoint X="2169849.784" Y="13731099.981" />
  <CutLinePoint X="2169849.56" Y="13731099.981" />
  <CutLinePoint X="2169727.495" Y="13731128.597" />
</CutLine>
- <WaterElevations>
  <WaterElevation ProfileID="PF_1" Value="747.6318" />
</WaterElevations>
```

Figure 4.36 XML File with Tag Definitions for Water Resources

The SDF2XML script tool converts the HEC-RAS SDF file into XML format, which represents an easier and more generic format for moving data between applications, in this case to import the datasets into an Arc Hydro geodatabase. As opposed to ASCII files, reading an XML file poses a parsing problem that can be programmatically addressed by means of various object models. The use of object models enables XML content to be presented to mainstream programming languages as a collection of objects with corresponding properties and methods without requiring a specially formatted read statement.

The two most well-known models for representing XML programmatically (to enable efficient document navigation) are the Document Object Model (DOM) and the Simple API for XML (SAX). By using these models, the hierarchical tree of data in the XML document can be parsed recursively to retrieve data, data can be accessed by tag name, and a list of nodes (children of an element) can be retrieved. All of these object-oriented capabilities can also be reversed to accomplish population of XML documents with new data. Thus, the object models can be used not only for deciphering XML documents but also to build them.

The SDF2XML application uses the DOM model to load the entire XML document into memory giving the flexibility to access any section of the document at any point in the application. To access a given section of the XML document, the DOM parsing tools are used to instantiate objects that point to the desired section and extract the information content by means of the object's methods and properties. The XML objects that need to be represented by the DOM model are the elements, attributes, and nodes.

In summary, the main advantage of an XML document is that it can be treated as a collection of objects and navigated as such instead of having to use traditional and tedious text parsing methods. Also, the self-describing nature of the data in XML provides the data context and understanding needed for application integration exercises like the one attempted hereby.

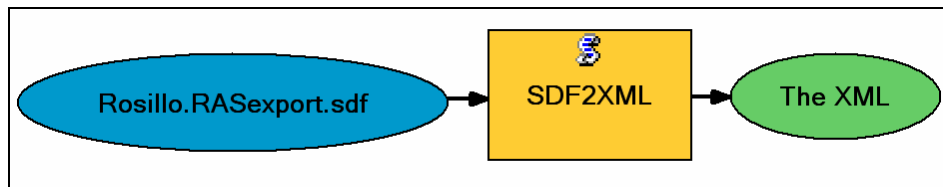


Figure 4.37 SDF2XML Model Component

4.1.12 The XML2GDB Script Tool

The XML2GDB application accomplishes the transfer of relevant output from the hydraulic simulation to the Arc Hydro geodatabase. In particular, the water surface elevation values for each cross section and the extent of the water surface in each of the cross sections. Figure 4.38 illustrates the hydraulic output extracted from XML content for the cross section “Rosillo, Upper, 96343.20”. The information delimited by the tags <WaterElevations> and <WaterSurfaceExtents> needs to be transferred to the geodatabase to proceed with the generation of the flood inundation maps.

```

- <CrossSection>
  <StreamID Value="Rosillo" />
  <ReachID Value="Upper" />
  <Station Value="96343.20" />
- <CutLine>
  <CutLinePoint X="2170685.584" Y="13731419.11" />
  <CutLinePoint X="2169520.583" Y="13731540.464" />
</CutLine>
- <WaterElevations>
  <WaterElevation ProfileID="PF_1" Value="751.604" />
</WaterElevations>
- <WaterSurfaceExtents>
  <WSEExtentPoint LeftX="2170124.19" LeftY="13731477.59" RightX="2170096.83" RightY="13731480.44" />
</WaterSurfaceExtents>
- <BankPositions>
  <LeftBank Value="547.32" />
  <RightBank Value="754.84" />
</BankPositions>
</CrossSection>
  
```

Figure 4.38 HEC-RAS Output in XML Content

The water surface elevations obtained from the HEC-RAS simulation for the cross section “Rosillo, Upper, 96343.20” in Figure 4.38 gets populated in CrossSection feature class and under the PF_1 profile elevation field (see Figure 4.39). In a similar manner, all the water surface elevations for the remaining cross sections in the hydraulic model are populated in the associated cross section table record. All the cross sections in HEC-RAS get a corresponding water surface elevation and they are identified in Arc Hydro by means of the RASCode whose components (Stream, Reach, and Station) are also present in the source XML file.

HydroID	STREAM_ID	REACH_ID	STATION	RASCode	PF_1
191	Rosillo	Upper	96343.2	Rosillo,Upper,96343.2	751.604
208	Rosillo	Upper	88421.27	Rosillo,Upper,88421.27	715.1536
237	Rosillo	Upper	79510.08	Rosillo,Upper,79510.08	679.9169
251	Rosillo	Upper	74267.24	Rosillo,Upper,74267.24	667.3485
268	Rosillo	Upper	64219.32	Rosillo,Upper,64219.32	645.9036
294	Rosillo	Upper	55649.9	Rosillo,Upper,55649.9	623.2077
316	Rosillo	Upper	45852.4	Rosillo,Upper,45852.4	605.7773
323	Rosillo	Upper	43789.5	Rosillo,Upper,43789.5	603.4001
354	Rosillo	Upper	28769.03	Rosillo,Upper,28769.03	579.0384
413	Rosillo	Upper	2.953	Rosillo,Upper,2.953	516.8511

Record: 0 Show: All Selected Records (10 out of 223)

Figure 4.39 HEC-RAS Output in Arc Hydro

Thus, the XML2GDB application uses the XML file created from the HEC-RAS output SDF file to update the attributes in the CrossSection feature class that describes water surface elevation. The name of the field storing the water surface elevation matches the name of a HEC-RAS flow profile in the XML file. If the field name does not exist in the feature class, it is created by the

XML2GDB tool at runtime. Figure 4.40 shows the schematic of this application under the ModelBuilder workflow.

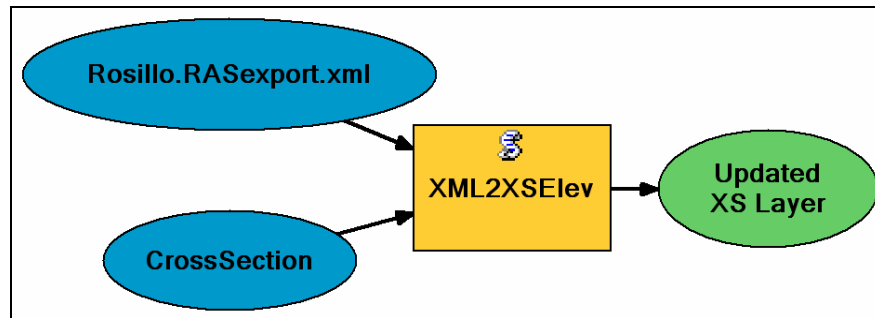


Figure 4.40 XML2XSElev Model Component

From the programmatic standpoint, the application first gets a hold of the source XML document and navigates through it to extract the flow profile names used for the hydraulic simulation in HEC-RAS. All the profile names found are stored in a collection of profile names called “colProfiles” for later reference in the code. Once the XML2XSElev application knows the outcome flow profiles it adds the profile elevation fields to the cross section feature class.

Next, the application reads the cross section feature class and loops through each cross section features extracting the RASCode “triplet” (a string made up of the combination of StreamID, ReachID, and Station as shown in Figure 4.39) identifying it. After extracting the RASCode for each cross section, the application navigates the XML file (using the DOM as an object oriented parsing tool) and gets the water surface elevation value for the current cross section RASCode and for each profile.

Once the water surface elevations for the current cross section are extracted they get stored into a collection of elevations (colElevs). Once the application has extracted the elevations for the current cross section feature it loops through the previously created collection of profiles (colProfiles) and writes the elevation values in the corresponding field matching the profile name in the collection of profiles. This process is repeated for each cross section feature in the Arc Hydro geodatabase.

The application then finishes execution by doing some standard programming tasks like closing the edit session for the workspace object representing the Arc Hydro geodatabase, saves the edits (new added profile fields and populated elevation values) and flushes the memory space being used by all the instantiated objects in the program.

In summary, the geospatial outputs from HEC-RAS include cross sections (2D cut lines), with water surface elevations for one or several profiles, and water surface bounds for each reach. The output cross section data can be read into a GIS layer and continuous water surface elevations between cross sections obtained by some type of spatial interpolation. Once all the necessary information is contained in GIS format, inundated areas can be mapped by comparing the interpolated water surface with the ground surface. To accomplish this, a series of geoprocessing steps needs to be undertaken which can now be wired together by means of the ArcGIS ModelBuilder. The next section describes each step (geoprocess) individually and the possible alternatives that can be used to achieve the same goals.

4.1.13 Spatial Manipulations of Hydraulic Modeling Results

After having the hydraulic modeling results inside the Arc Hydro geodatabase (CrossSection feature class attributed with profile elevations) it is possible to take advantage of all the geoprocessing capabilities of a geographic information system to generate a realistic and georeferenced flood inundation map representing the flow conditions of a given flow profile. Traditionally, GIS systems have been empowered with “spatial analyst” type of components that allow one to create, query, map, and analyze cell-based raster data and to perform integrated vector–raster analysis. Some of the typical geo-tasks normally accomplished through the spatial analyst component are the conversion of feature themes into grid themes; create continuous surfaces from line features or from scattered point features, create contour, slope, and aspect maps out of the continuous surfaces, perform cell-based map analysis, perform Boolean queries and algebraic calculations on multiple grid themes simultaneously, perform neighborhood and zone analysis, and perform grid classifications.

All of these spatial tasks can now be integrated to create a workflow of geoprocesses needed to accomplish a final goal. For building a flood inundation map using the aforementioned spatial analyst capabilities in GIS the next basic steps need to be accomplished:

Generate a suitable terrain surface model (Digital Elevation Model, DEM), that is, a cell-based continuous surface of ground elevations in grid/raster format.

Generate a suitable water surface model, represented by a cell-based continuous surface of water elevations in grid/raster format.

Subtract the land surface elevation from the water surface elevation to obtain a continuous surface of water depths.

These steps require a much more elaborate sequence of geoprocesses already available in GIS.

For the Map2Map application, the terrain surface model is considered a given input that does not need to be further processed inside the Map2Map application. Contours are a common source of detailed digital elevation data for local and regional studies. For Rosillo creek, the digital elevation model (ground surface model) was created from survey field studies that generated 2-foot contour line maps in the form of CAD files (.dwg, dgn, dxf). Spatial analyst functionalities were used to convert the contour line map into a TIN (Triangulated Irregular Network) and the TIN in turn is used to generate a DEM (raster representation on the surveyed terrain elevations).

Even though national databases (like the USGS National Elevation Dataset, NED) are available through the internet and supply digital elevation models for the United States, the resolution of these datasets is too coarse (up to 10 m resolution based on 24K scale maps) for use in flood studies which require a more detailed representation of the terrain model (less than 10 m resolution) for the river's main channel and lateral overbank sections as well as the inclusion of relevant soft and hard break lines in addition to mass points for the representation of infrastructure (like buildings and roads footprints) over the floodplain. Figure

4.41 illustrates a 5-foot contour line map over the Rosillo creek study area generated from a survey study developed by the city of San Antonio.

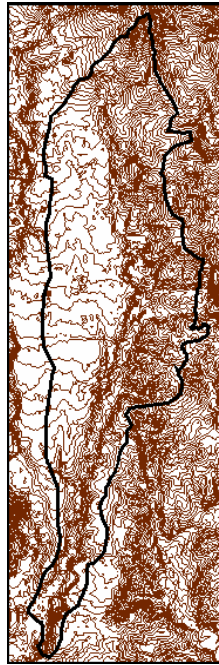


Figure 4.41 Terrain Model of Rosillo Creek Represented by 5 Foot-Contours

A detailed land survey study (developed by contractors of the City of San Antonio) provide a 2-foot contour map covering the extent of the Rosillo creek's floodplain. The topographic information from this land survey was also used to supply the cross section geometric information needed for the configuration of the HEC-RAS hydraulic model. Figure 4.42 shows the 2-foot contour line map over the entire floodplain of Rosillo creek and an inset for a lower section of the creek.

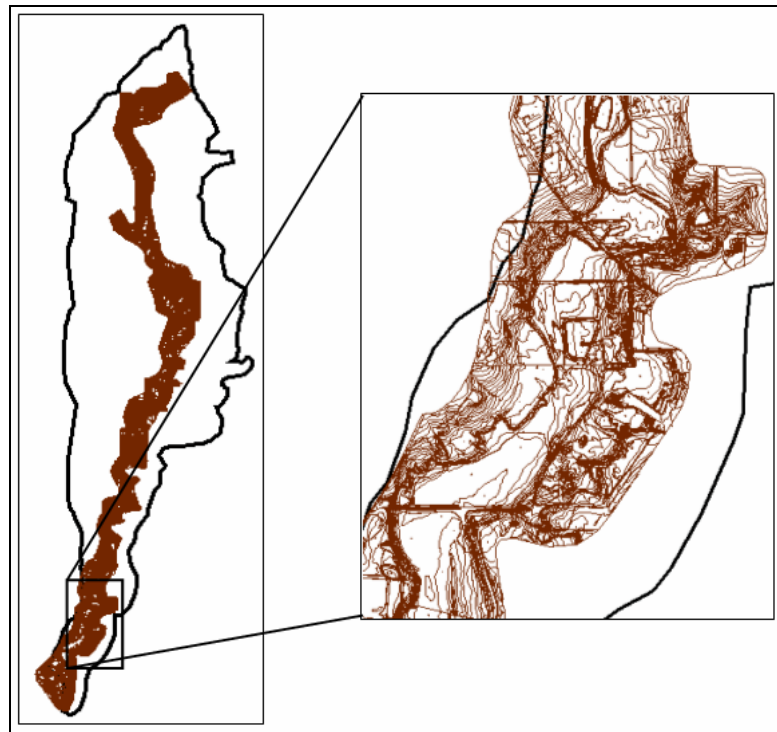


Figure 4.42 2 Foot-Contour Lines for Rosillo Creek's Floodplain

A contour line map can be used to create a TIN (Triangular Irregular Network) based on standard GIS spatial analyst functionality. A TIN model represents a surface as a set of contiguous, non-overlapping triangles made from a set of points called mass points with the surface between each triangle being represented by a plane. For a TIN to be a good representation of the surface, mass points must be strategically located where there is a major change in the shape of the surface (the peak of a mountain, the floor of a valley, and the top and bottom edges of cliffs). The main advantages of the TIN model over the GRID model are the ability to describe the surface at different levels of resolutions and the consequent efficiency in storing the data. TIN models are not as readily available

as GRID models and not as easy to store and manipulate (GRID models can be represented by ASCII files) and the derived terrain features do not have a smooth and natural appearance as do GRID models.

A TIN model is normally the first continuous surface generated out of the contours because all the vertices of the contour lines can be used as mass points for triangulation and because of the ability to include break lines in the model definition to honor physical features like rivers and roads. By using all vertices for building the triangulation, the continuous surface will not only honor the elevation values at every known location but also provide a higher level of triangulation over high relief areas (more and closer contours) than over low relief areas (few and spaced contours). This ability for generating surfaces that honor all point values with various levels of triangulation makes the TIN model more attractive as the first terrain model to be created from contours. Figure 4.43(a) depicts the TIN model for Rosillo creek generated from all the vertices of the 2-foot contour lines shown above. The extent of the TIN model goes beyond the extent of the main stem floodplain due to triangles generated between border points.

Given that the delineation of the flood inundation map is based on raster products (intersection with a grid of water surface elevations), after the TIN model is generated for the terrain elevation, a GRID model with a 20 foot cell size was generated using standard spatial analyst functionalities and clipping the created grid to just the extent of the floodplain under analysis. Figure 4.43(b) shows the final GRID model for the ground surface elevation.

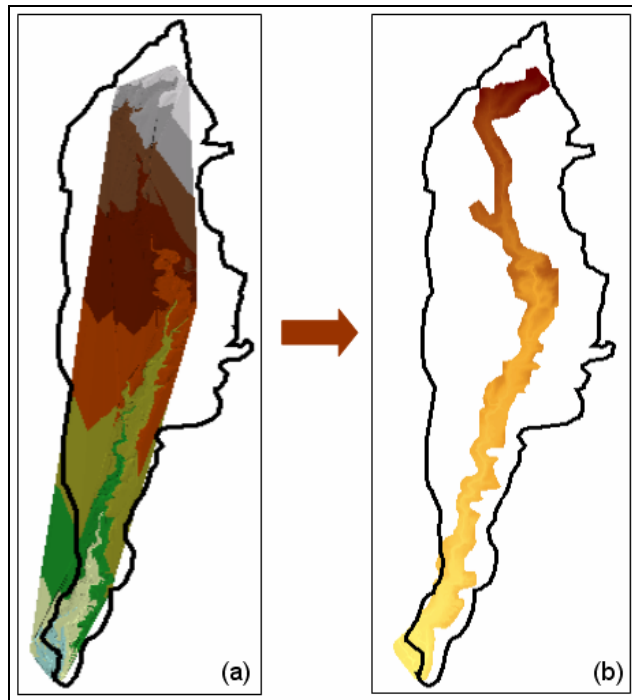


Figure 4.43 Rosillo Creek's terrain model as TIN (a) and GRID (b) formats

Therefore, the digital elevation model was pre-processed outside the Map2Map application and the GRID model provided as an input to the application for further manipulation and generation of the flood inundation maps. The following sections describe the components dealing with the flood delineation phase.

Given that the terrain surface elevation model is given to the Map2Map application and it is not affected by the hydrologic and hydraulic simulations integrated by it, the only surface model that needs to be created at run time is the water surface elevation model.

The main objective of this Map2Map component is the generation of a comprehensive sequence of steps for subtracting the terrain elevation surface from the water surface elevation to obtain water depths and to produce an accurate representation of the floodplain inundation map as schematically shown in Figure 4.44.

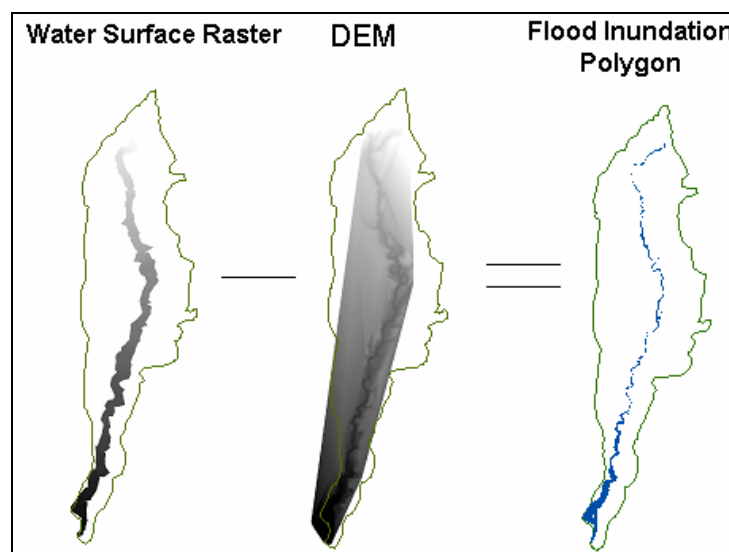


Figure 4.44 Flood Map Created by Subtracting Land Surface from Water Surface

4.1.13.1 Create and Edit TIN

The “Create TIN” tool is a standard ArcGIS tool, which is used to create a new TIN with an extent provided by an existing input dataset. In principle, this tool can be executed at any point in time but for the Map2Map workflow it was set up to execute only after the cross sections have been successfully updated (populated with associated water elevations) from the previous XML2GDB process. Once the “blank” TIN surface is created, the “Edit TIN” tool (another

standard ArcGIS tool), is used in Map2Map to add the cross sections as soft break lines (no variation in water elevation along them), using the field name of the flow profile (e.g., PF_1) populated with the water surface elevations along the cross section cut lines. The final outcome is a TIN surface of water elevations with triangle nodes lying over the cut line vertices of the hydraulically defined cross sections.

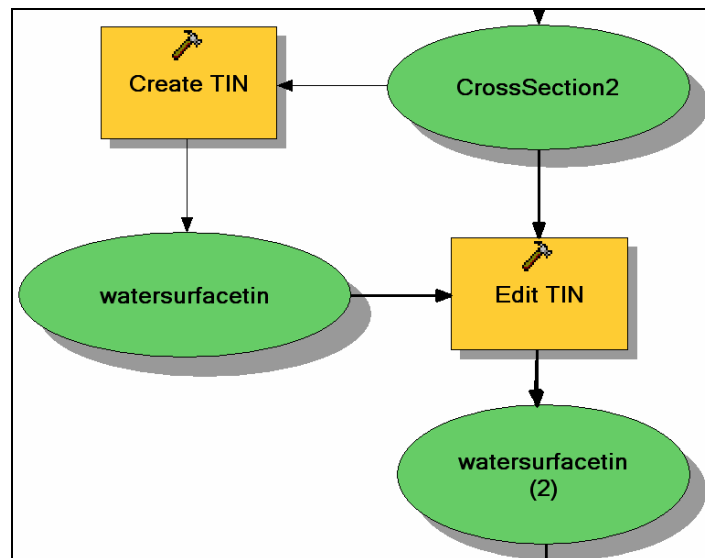


Figure 4.45 Create and Edit TIN Model Component

TIN to Raster

The TIN to Raster is another standard ArcGIS tool (represented by a hammer icon). It is used to convert the water surface TIN to a raster dataset, so that it can be readily compared to a raster surface representing land surface elevation (DEM).

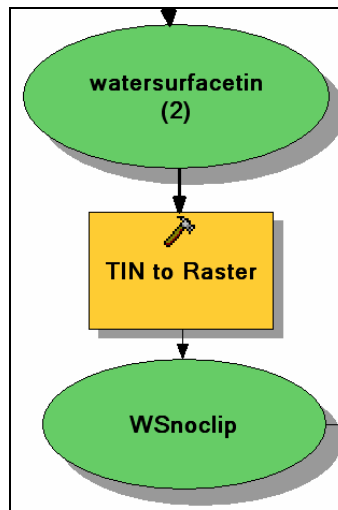


Figure 4.46 TIN to Raster Model Component

4.1.13.2 Extract by Mask

Before the depth of water in the floodplain can be mapped, the raster surface of water elevations must be made consistent (same cell size and spatial extent) with the existing DTM of the floodplain under analysis. Due to the bounded extent of the first generated TIN of water surfaces, the raster obtained from the previous process also goes beyond the spatial extent of the floodplain and interpolates water surface elevation values over outlying areas that are not accurate or reliable (areas of non existing contour lines). To avoid this, the water surface raster is clipped to the extent of the floodplain (main channel plus lateral overbanks) being described by using a mask (a unit value raster) with the desired spatial extent, the Boundary layer of the floodplain under study. This layer is part of the pre-processing phase and may consist of a polygon obtained by joining the end edges of all the HEC-RAS cross sections.

By doing this, the water surface elevations assigned to each cross section can be represented by a continuous surface having the same cell size and spatial extent of the DEM. The ModelBuilder schematic for clipping the water surface grid based on the “Extract by Mask” standard tool is shown in Figure 4.47.

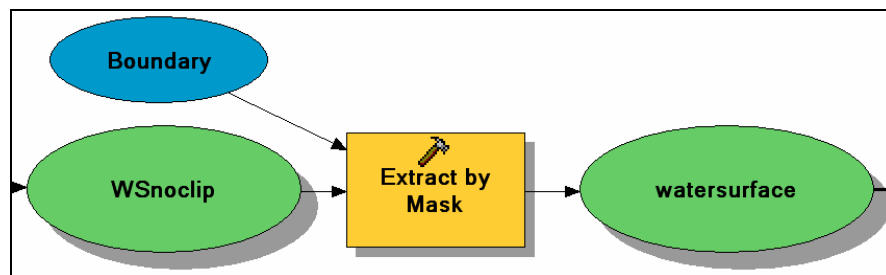


Figure 4.47 Extract by Mask Geoprocess Component

Up to this point, the Cross Section layer in Map2Map, populated with hydraulic water surface elevations from HEC-RAS, has been used to generate a grid model of water surface elevations that is consistent (same extent and resolution) with the terrain elevation grid model. The concept of this transformation is shown in Figure 4.48.

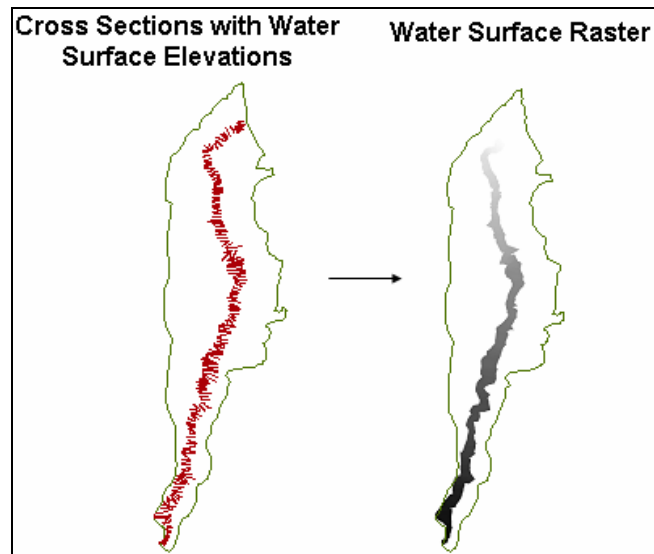


Figure 4.48 Continuous Water Elevation Surface from Elevations at Cross Sections

The next phase in the raster processing component of Map2Map is in charge of generating a vector product depicting the flooded areas associated with the hydrologic event fed into the application.

4.1.13.3 Create Flood Polygon from Surfaces

The “Create Flood Polygon from Surfaces” model tool compares (a subtraction operation) the land surface elevation raster with the water surface elevation raster to produce a polygon representing inundated area (water depth).

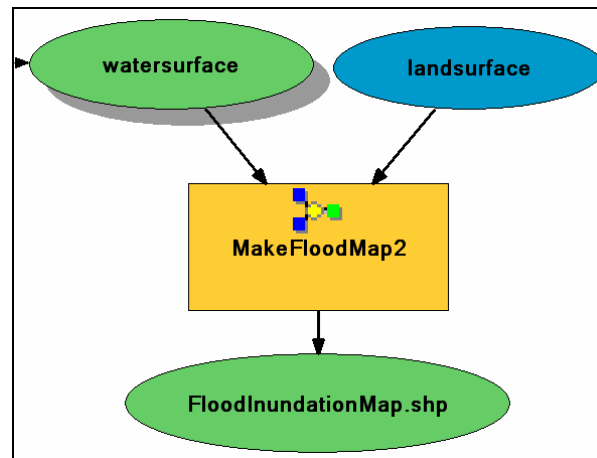


Figure 4.49 Create Flood Polygon Model Component

To accomplish this raster to vector transformation, various approaches can be adopted based on the multiple standard geoprocessing tools provided in ArcGIS 9.0. Each alternative consists of different combinations and sequences of the various available ArcGIS tools. Two of these options are shown here for illustration, but every modeler may come up with alternative approaches to obtain comparable outcomes. Figure 4.50 shows the workflow layout for this conversion and the discussion for each involved process (hammer rectangles) follows.

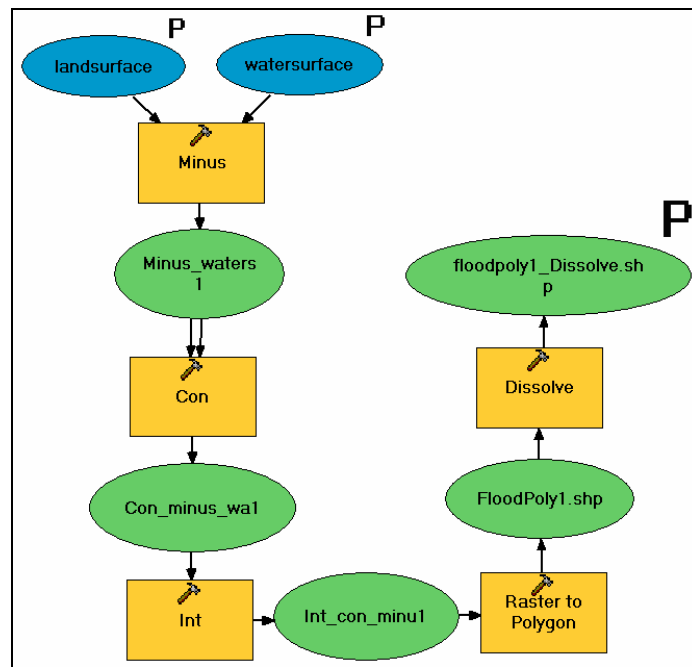


Figure 4.50 Create Flood Inundation Map Model Workflow (Option 1)

Minus operation

A map algebra operation function is performed to subtract cell values in the terrain surface raster from the cell values in the water surface raster. This operation contains positive values representing cells that are inundated and negative values without any physical meaning.

Con Operation

The Conditional raster operation is used to filter the meaningless negative and non-inundated zero values of the previous map algebra operation and generate an all positive value raster having water depth cell values for the floodplain under analysis.

Int Operation

The Integer operation is needed to generate an integer value raster to enable the implementation of the “Raster to Polygon” tool. This tool does not affect the level of precision of the final result since the required output (at least for the current implementation) is represented by a single polygon of inundated area that does not contain any internal discretization of water depth values in the floodplain.

Raster to Polygon Operation

The Raster to Polygon tool transforms the water depth raster surface into a gridded polygon corresponding to the vector representation of the water depth raster. The outcome of this tool is a polygon feature composed of multiple square polygons representing the raster cells and having the water depth values populated on each polygon.

Dissolve Operation

The Dissolve function dissolves the multiple inundated polygons to produce a single polygon representing the entire inundated area of the floodplain under study with no discrimination for water depth values (just water depths greater than zero).

Another alternative to accomplish the floodplain delineation consists of replacing the final Dissolve geoprocessing operation with the Reclassify operation occurring prior to the “Raster to Polygon” geoprocess.

The “Reclassify” operation is used to reclassify the class value intervals of the integer water depth raster with a single common value for the whole surface

and then use it as the field value of the Raster to Polygon operation. The final raster outcome should be comparable with the previous raster and the duration of this alternative is also comparable. The schematic of this second alternative approach in ModelBuilder is shown in Figure 4.51.

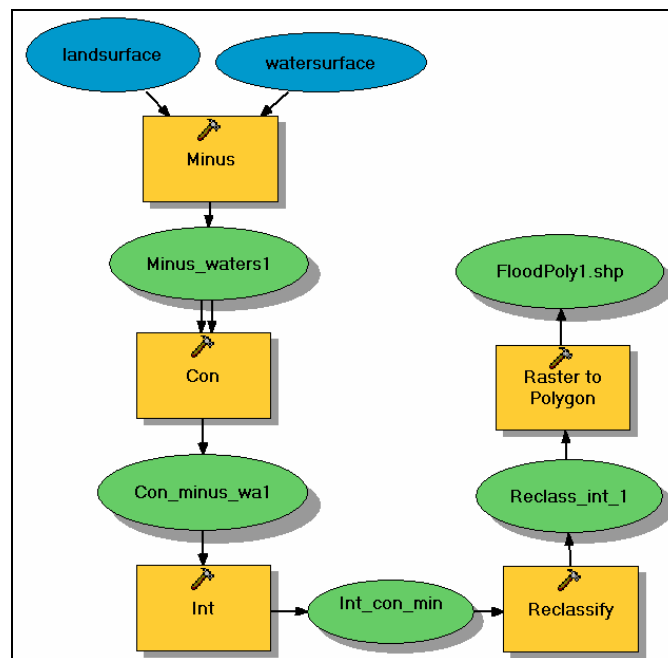


Figure 4.51 Create Flood Inundation Map Model Workflow (Option 2)

Therefore, the modeler may try different alternatives for the delineation of the floodplain map to make sure the final scheme satisfies calibration, and validation criteria.

Even though the above discussion about each of the individual components of the Map2Map application is for the case of historical precipitation events from NEXRAD sources, most of the individual processes can be reused for other use cases due to the modular character of the application. The second use

case that was developed for Map2Map is the use case of Design Storm Events as the source of precipitation. The “historical event” use case is mostly geared towards automated floodplain mapping of all the range of possible realizations of precipitation. This capability can be exploited in the generation of “design floodplains” (avoiding classical assumptions of rainfall and runoff with same probability distributions) and to readily generate rating curves (under new development scenarios) at any desired ungauged locations and for flood impact analysis studies. The historical event use case can also be easily extended to provide necessary groundwork for near real time flood forecasting.

4.2 MODEL COMPONENTS FOR THE DESIGN STORM USE CASE

The “design storm” use case is mostly geared towards planning and management for the purpose of structural designs, regulatory floodplain map updates, and prioritization of projects. For the current Map2Map application, an alternative for the design storm use case was developed such that it reuses most of the process components in the historical event use case. Two new processes had to be developed for the design storm use case which replaced the first three processes of the historic event use case. Figure 4.52 depicts the entire Map2Map workflow for the Design Storm use case starting at the upper left yellow box (DStormSeries process) end and ending at the lower right end (Create Flood Polygon from Surfaces process model).

As mentioned above, the complete workflow for the design storm use case starts with the Sub-Model “DesignStormSeries” and follows with the user-defined

script DesignStorm2HMS thereafter, it reuses and proceeds with the same tasks as in the historic event use case.

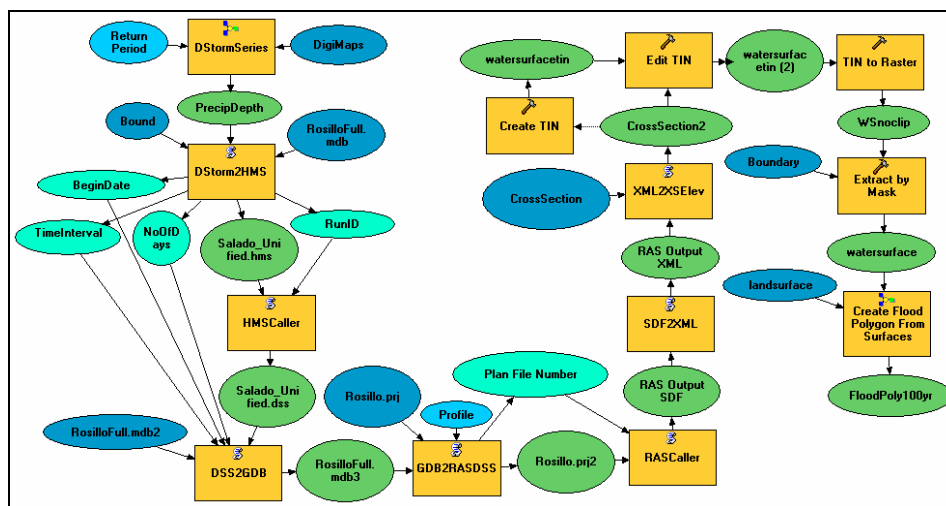


Figure 4.52 Map2Map Workflow in ModelBuilder for Design Storm Use Case

The determination of a suitable rainfall event to be used in water resources planning and design projects is one of the most relevant factors to be considered. The standard approach (agency-approved) is the use of design storms whose characteristics are commensurate with the importance of the study project at hand. To obtain a given design storm, statistical frequency analysis is performed over the entire history of available point precipitation records. For each year of historical record, annual maxima are extracted from standard storm durations and histograms constructed using probability models to obtain maximum values for a given duration and precipitation depths associated with standard frequencies (return periods). After rain depths are found for specific frequencies and durations at point gage locations, areal precipitation might be inferred by drawing iso-rain

contours (isohyets) or by interpolating point values to generate iso-surfaces representing precipitation values for a given frequency and storm duration.

The execution of the aforementioned procedure over a region produces a collection of synthetic precipitation events that can be used for regulatory purposes and to promote standard and comparable engineering practices. These types of studies have been developed over the U.S. by the National Weather Service and used as a national reference. These studies have been presented in the form of reports that serve as the standard in the U.S. The U.S. Weather Bureau technical paper no. 40, also known as NWS TP-40, for durations from 30 minutes to 24 hours and developed by Hershfield in 1961. The need for inclusion of bigger storm durations prompted supplemental studies in 1964 that included storm durations for 2 days to 10 days, reported as NWS TP-49. The need for smaller storm durations of 30 minutes or less (in urban drainage design or storm sewer design) prompted another study by the National Weather Service in 1977 commonly known as Hydro-35 for storm events having durations from 5 minutes to 60 minutes.

A digital rainfall atlas for Texas (containing raster isohyetal maps of design rainfall depth for various event durations and for standard return periods) was made available for this study by Halff Associates, Inc. based on the three reports mentioned above (TP-40, TP-49, Hydro-35). This digital atlas contains raster isohyetal maps for storm durations of 5 minutes, 15 minutes, 30 minutes, 60 minutes, 2 hours, 3 hours, 6 hours, 12 hours, 24 hours, 2 days, and 4 days. For each of the listed durations the series of frequency values corresponds to 1-year,

2-year, 5-year, 10-year, 25-year, 50-year, 100-year, and 500-year return period. Figure 4.53 shows three grid cell isohyetal maps for the 100 year (1%) exceedance frequency and storm durations of 6 hours, 12 hours, and 24 hours as developed by Halff Associates.

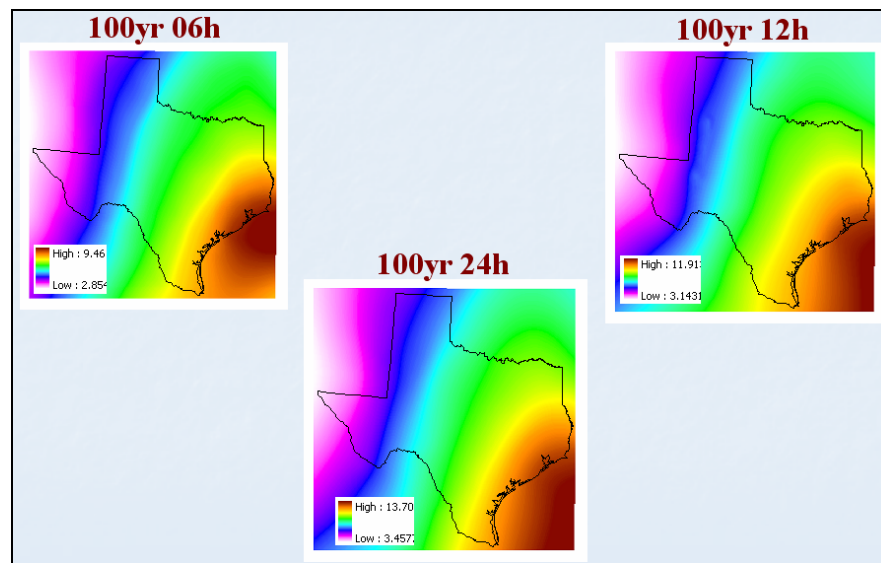


Figure 4.53 Digital Design Rainfall Maps for Texas (color scale values in inches)

There are four options available in HEC-HMS for generating a synthetic precipitation for each subbasin: the frequency storm, the standard project storm, the SCS hypothetical storm, and the user-specified hyetograph. The frequency storm method uses depth-duration data usually obtained from the standard publications mentioned above i.e., TP-40, TP-49, and Hydro-35.

The frequency storm method uses statistical data to generate balanced storms (peak intensity at midpoint) with a specific exceedance probability. The frequency storm method was selected for the Map2Map implementation of the

design storm input given its possibility of direct ingestion of Depth-Duration input data from the aforementioned publications and to be compliant with the most common engineering practices.

4.2.1 The DesignStormSeries Model

For a selected frequency storm the method requires the precipitation depth for a series of durations starting from the duration of the maximum intensity and continuing up to the total storm duration. In order to obtain the depth for various durations, geoprocessing tasks can be used to query the collection of digital design storms in raster format and extract the depth for the needed duration and for the selected frequency. This can be done in using ArcGIS ModelBuilder by intersecting the study area with the raster of precipitation depths and employing zonal statistics to get an average precipitation depth value over the watershed.

Figure 4.54 illustrates the extended Arc Hydro table used to store the content of this spatial query for precipitation depths of various durations. The MEAN field contains the mean precipitation depth on the area of interest extracted with Zonal Statistics in ArcGIS. The Frequency field contains the return period (100 year) and the Duration field contains the series of durations needed to generate the synthetic storms (15 minutes, 30 minutes, 60 minutes, 2 hours, 3 hours, 6 hours, 12 hours, 24 hours, 2 days, and 4 days).

AREA	Frequency	Duration	MEAN
80000000	100y	15m	1.91676
80000000	100y	60m	4.23873
80000000	100y	02h	5.33947
80000000	100y	03h	5.91161
80000000	100y	06h	7.18129
80000000	100y	12h	8.62883
80000000	100y	24h	9.94795
80000000	100y	02d	11.529
80000000	100y	04d	13.1891

Figure 4.54 Depth-Duration-Frequency Table in Arc Hydro

The above table is generated in ModelBuilder by a model that links together all the script and standard tools needed sequentially perform Zonal Statistics over the series of digital rainfall maps and to produce the above table. Figure 4.55 shows a snippet of the ModelBuilder workflow of the “DesingStormSeries” Model tool generated for this purpose. The model takes the selected return period and the location of the raster digital rain maps as the main variable inputs. It starts populating the PrecipDepth table by performing zonal statistics on the first duration (15 minutes) raster and for the 100 year frequency (g100y15m) and appending the results to the PrecipDepth table. After the first duration raster of rain depths is processed the next duration raster in line (g100y60m) is processed and its corresponding precipitation depths appended to the PrecipDepht table. This process is repeated for all durations of the selected storm frequency.

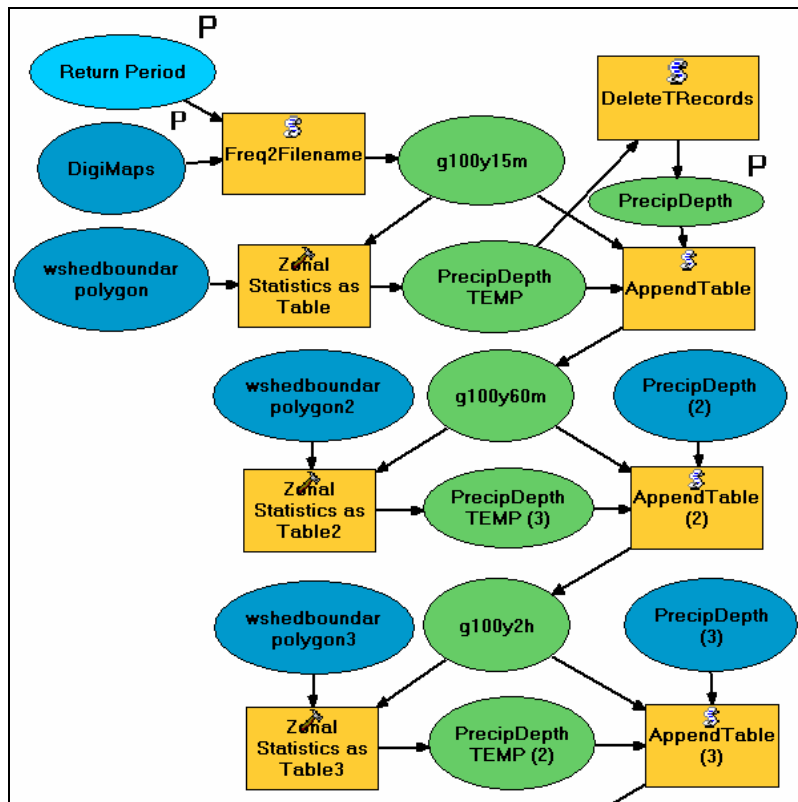


Figure 4.55 Workflow for Extracting Rain Depths from Digital Design Storms

4.2.2 The DigitalStorm2HMS Script Tool

After the Depth-Duration-Frequency function is established and stored in Arc Hydro for the selected annual exceedance probability the appropriate input files for HEC-HMS can be generated. The main objective of this application is to write the meteorologic file for the frequency storm method in HEC-HMS.

The DigitalStorm2HMS application starts by reading the previously created PrecipitationDepth table in Arc Hydro to extract the mean precipitation depth value for each duration and for the selected storm frequency. Then, the DigitalStorm2HMS application reads the Watershed feature class to extract the

basin's area to assign the storm size parameter. The Watershed feature class is also used to extract all the subbasins' names for the meteorologic file which serve as the recipients of the synthetic storm information.

Once the application has all the needed elements, it writes the ASCII meteorologic file (.met file extension) for HEC-HMS describing the precipitation-depth function of the given frequency. Figure 4.56 shows the ASCII file section that is written by the application for a 1% frequency event.

```
Method Parameters: Frequency Based Hypothetical
  Exceedence Frequency: 1
  Single Hypothetical Storm Size: Yes
  Convert to Annual Series: No
  Storm Size: 0
  Total Duration: 1440
  Time Interval: 5
  Percent of Duration Before Peak Rainfall: 50
  Depth: 0.87
  Depth: 1.91
  Depth: 4.25
  Depth: 5.57
  Depth: 6.23
  Depth: 7.13
  Depth: 8.05
  Depth: 9.91
End:
```

Figure 4.56 Section of Meteorologic file for Frequency Storm Method

After the DigitalStorm2HMS application writes the meteorologic file it has to be registered in various places. The HMS project file (.hms file extension) is the first place the meteorologic file needs to be registered. The DigitalStorm2HMS application then accesses the project file and looks for a registry of the new met file, if the registry is not found a new registry is created

for it. Finally, the DigitalStorm2HMS tool generates a control file (.control file extension) to configure a simulation time window that should span the entire rise and fall of the runoff hydrographs to be generated based on the fed synthetic storm by adding the basin's travel time to the total storm duration.

The ModelBuilder schematic for the DesignStorm2HMS script tool is shown in Figure 4.58.

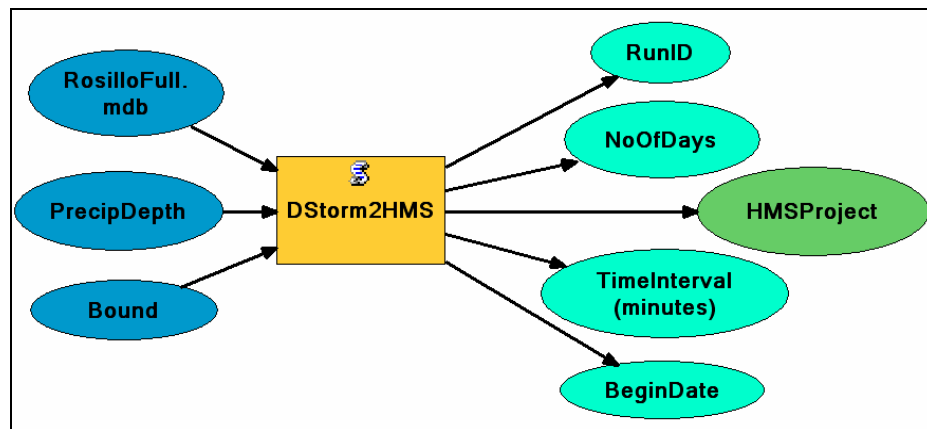


Figure 4.57 ModelBuilder Schematic for DesignStorm2HMS script tool

Chapter 5 Conclusions and Recommendations

5.1 A SEAMLESS END-TO-END GEOGRAPHIC INTEGRATION

The main scientific contribution of this research corresponds to the connectivity of two independent and stand-alone modeling systems (HEC-HMS and HEC-RAS) under a map-centric framework for the purpose of delineating flood inundation maps and streamlining the process to accomplish the integration. By facilitating integration of models and applications it is possible to take advantage of well-tested and approved design concepts, computer code and model components across multiple target-specific engineering models. This research provides a sound approach for “bringing the models together” under a map-centric framework for automated and sequential hydrologic and hydraulic simulations.

GIS was employed in a dual fashion, for the spatial identification of exchange points of information between the models and to automate the workflow of all involved processes. In contrast with the regular piecewise approach this integration avoids time-consuming, error prone, and more expensive flood study developments under a comprehensive spatial framework. The spatial character of the needed points of information exchange between the involved models makes this approach consistent and robust.

The system facilitates the coupling of GIS and engineering models by implementing a geographic integration based on existing and new relationship classes (spatial associations) on top of a standard data model, the Arc Hydro data

model for water resources (Maidment, 2002). This spatially oriented system for the linkage of hydrologic and hydraulic modeling systems is based on geographic representations of the stream geometric network, schematic network, and model-centric features that have been proved feasible and extensible to even more ambitious implementations. By following the proposed integration methodology, other hydrologic information systems so defined may serve as a reference frame to link the operation of other supplemental engineering/environmental models such as potential water quality components (e.g., QUAL2E, HSPF, SWAT, etc.).

In linking HEC-HMS and HEC-RAS through the GIS-based Arc Hydro data model (Maidment, 2002), the inherent expertise of each model is exploited by allowing one model to determine streamflow hydrographs, and the second to determine water surface elevations and flood extents as a function of given input inherited from sequential simulations. By doing this, a traditionally tedious, time consuming, and error prone modeling task has been streamlined and can now be accomplished under a more comprehensive and controlled environment that takes full advantage of the spatial character of the study area undergoing the modeling exercise which executes in about four minutes what could normally take hours of work.

The implementation of a standard data model for water resources, Arc Hydro, the compliance of external and specialized stand-alone modeling systems with standard interconnection protocols (COM technologies), and the higher level of customization provided by the new ArcGIS 9 software provides all the

ingredients and a suitable platform for gluing together heterogeneous modeling resources under a convenient geographic framework.

5.2 MODEL CODE ELEMENTS AS GEOGRAPHIC INTEGRATORS

The geographic integration uses Arc Hydro to georeference information exchange points represented by HydroJunction and CrossSection features in the spatial context. The georeferenced HydroJunction points get associated with hydrologic elements in HEC-HMS by means of the SchematicNetwork (an existing topologic feature in Arc Hydro) which must accurately resemble the topology of the hydrologic model.

Arc Hydro identifies those HydroJunctions that carry runoff hydrographs generated in HEC-HMS by selecting those HydroJunctions associated with a SchematicNode serving as a runoff entry point into the main stem. This association is represented by the HydroJunctionHasSchematicNode relationship built inside Arc Hydro. Through the relationship class, the Map2Map application reaches the HMS predefined names for the hydrologic elements that are passing runoff information. These names are model codes termed HMSCodes for HEC-HMS which are populated in the SchematicNetwork to properly identify the hydrologic elements inside the HEC-DSS time series file holding the runoff information that needs to be transferred.

By doing this, the HydroJunctions carrying runoff hydrographs are identified. Next, the HydroJunctions need to identify the next downstream cross section in HEC-RAS topology to receive the runoff hydrograph. This identification is done in Arc Hydro by means of a new relationship class in Arc

Hydro, the HydroJunctionHasCrossSection relationship class built inside Arc Hydro.

Through this relationship class, the Map2Map application reaches the RAS predefined names for the hydraulic elements (cross sections) that are receiving the runoff information. These predefined names are again model codes termed RASCodes for HEC-RAS which are populated in the CrossSection feature class to properly identify the hydraulic elements in HEC-RAS configuration receiving the runoff information that needs to be processed. The cross section elements and their names are used in the Map2Map application to generate an appropriate flow file listing the cross sections that receive runoff data and describing the source location of that data, that is, references to HEC-DSS pathnames identified by means of the previously inferred HMSCodes.

Thus, the georeferenced CrossSection features inherit runoff data by means of a relationship class that associates them with the HydroJunction exchange points which in turn must follow the HEC-HMS topology replicated in the SchematicNetwork in Arc Hydro representing the hydrologic model component.

A spatial integration so defined enables the integration of all type of external applications using the geodatabase as the common geographic framework which can be readily updated to reflect new developments in the basin.

5.3 RAINFALL INPUTS OF VARIOUS FORMS

Taking advantage of the modular and flexible character of the integration, the developed system provides a suitable framework for the analysis of multiple rainfall inputs: historic, real time, forecast, and synthetic.

The developed floodplain modeling system for single rainfall events is suitable for historic, real-time, and forecasted flood impact analysis. In other words, the system can perform simulations based on real event realizations and not only simulations of synthetic design storm realizations used in conventional floodplain planning and management purposes.

Additional functionalities to loop through the system may be used to also support annual-based and project life-based impact analysis by evaluating the floodplain response to a series of possible events (of a given duration) and computing frequency values from the entire sample range.

Thus, the current version of the development is easily adaptable for the ingestion of rainfall inputs of various forms. Historic storm events, real time rainfall events, precipitation forecasts, and synthetic or design precipitation inputs can all be accommodated to drive the integration system thanks to its modular structure and ability to build new workflow models that reused already existing code components. Thus, precipitation input datasets to drive the developed Map2Map integration might come in several formats depending on the agency, the means, and the spatial reference used for providing the data.

The proposed modular model integration provides the flexibility to ingest rainfall inputs of various forms without affecting the “core” of the processes

involved in the transformation of rainfall into flood maps. The modular character of the approach allows code reuse of generic tasks that remain constant across different rainfall scenarios and implementations and allows straight forward replacement of specialized components across the diverse implementations.

The modular system enables a number of associated benefits like code reuse, maintenance, and easy adaptability to other implementations. Custom designed process components represented by script tools can be easily replaced to handle different input formats or task alternatives. The current development allows the ingestion of various types of precipitation events: historic, real time, forecasted, and synthetic datasets.

Ingestion of historic rainfall events allows single event evaluations for planning and regulation against extreme events as well as for potential looping through the entire range of historic events of a given duration to facilitate frequency analysis studies. Ingestion of real time rainfall events provides support to improve flood warning and response systems.

Ingestion of forecasted rainfall events provides the promise of even better flood warning systems that could implement greater lead response times. This type of rainfall is not as reliable as actual recorded datasets and therefore the associated floodplain maps generated through the integration scheme might be taken with caution.

Finally, the ability to ingest standard design storms as digital maps provides support for flood hazard zoning and insurance programs like FEMA's Flood Insurance Rate Maps and National Flood Insurance Program. For instance,

floodplain maps obtained based on synthetic design storms may depict a series of flood hazard areas inside the floodplain boundaries based on the 1-percent-annual-chance flood event (100-year base flood elevations) as well as the delineation of the regulatory floodway by allowing fast and consistent encroachment analysis. Thus, the proposed integration might be used in multiple implementations like single-event target planning, flood warning systems, develop and update flood frequency studies, and standard design storm planning to update flood hazard zone subdivisions, floodplain and floodway delineations, base flood elevation analysis, etc.

5.4 EXCHANGE OF SPATIALLY-REFERENCED TIME SERIES

Also, critical to the success of this integrating effort is the proper transfer of spatially-referenced time series records between the linked model features and the wrapping geographic framework by means of application data bridges. For the current implementation, the transfer of time series records was done at three levels, between the Arc Hydro geodatabase and HEC-DSS system (for preparing modeling inputs), from the HEC-DSS system to the geodatabase (for receiving modeling outputs), within the Arc Hydro geodatabase (for performing spatial associations of the time series records to fulfill model's requirements).

The exchange of spatially referenced time series records between the Arc Hydro geodatabase and HEC-DSS allows the Map2Map application to generate input time series for the two HEC models (HMS and RAS) and empowers the integration by allowing the geographic framework to identify and select the features that possess modeling information that needs to be relayed to the next

process model in sequence. For the transfer of information towards HEC-DSS the Arc Hydro structure has been extended with an additional table in charge of storing the necessary HEC-DSS descriptors that characterized the time series records to be transferred. The integration of diverse time series representations is accomplished under a single and extended data model in Arc Hydro that allows time series transformations between NEXRAD, Arc Hydro, and HEC-DSS formats.

The exchange of time series records within the Arc Hydro geodatabase allows Map2Map to change the spatial-reference of the time series records. So, for instance, the starting precipitation records that are associated with NEXRAD cells in space need to be spatially referenced to subbasins describing the hydrologic domain in HEC-HMS and to generate the required hyetograph for each subbasin that will be transformed into runoff.

The exchange of spatially referenced time series records between HEC-DSS and the Arc Hydro geodatabase allows the Map2Map application to identify and select the hydrologic output from HEC-DSS that needs to be transferred for hydraulic input in HEC-RAS. Even though in principle the hydraulic setup might be directly referenced to the hydrologic input given their mutual ability to read HEC-DSS records, it was found that the generated runoff hydrographs in HEC-HMS may be stored in multiple HEC-DSS pathnames complicates their reference in HEC-RAS, which relies on a single DSS pathname to reference the streamflow inputs.

Besides this problem with the multiple pathname generation in HEC-DSS, it is also more convenient to have a complete HEC-RAS project in a single directory location that is fully self-contained and independent. For the aforementioned arguments, it was decided to filter the multiple DSS records in HMS to select only the needed time series records to obtain a consistent RAS configuration.

5.5 AN ENSEMBLE OF MODELS AND TOOLS

This research exposes the benefits of building a holistic system based on three types of models, process models, data models, and workflow models. The process models representing very specialized and stand-alone simulation models in charge of modeling physical processes, rainfall-runoff transformations and river channel streamflow routing in this case.

Two process models and an Arc Hydro data model for the study area are being executed as tools and referenced as source/target datasets together with customized and standard applications for the exchange of spatially referenced time series and the spatial delineation of the floodplain maps. All the previous components are ensemble on a workflow model represented by the ArcGIS ModelBuilder. By doing so, all the building blocks used in the conventional piecewise approach are being grout together to achieve a “seamless” integration.

It is demonstrated in this research that the use of ArcGIS Model Builder to automate the workflow sequence empowers a “GIS tool” to play the role or emulate of a full engineering model under a centrally managed geographic framework.

5.6 A TRUE SPATIAL APPROACH

Automated Flood map generation based on the true spatial distribution of the main variables is another key contribution in this research. By considering interpolated water elevations for the generation of a continuous water surface and a detailed terrain surface over the entire floodplain the obtained flood inundation maps are more consistent with the spatial configuration of the floodplain.

Thus, the developed floodplain modeling system can help provide distributed flood assessments as opposed to the traditional concept of aggregation to a damage index location that represents the hydrologic, hydraulic, and economic conditions of an entire floodplain section (regular lumped assessment approach). By doing this, the true spatial nature of the floodplain is considered and reflected in more accurate flood assessment studies.

5.7 A REGIONAL APPROACH

Most flood studies are developed at the local scale and by local agencies for the purpose of evaluating proposed projects with the potential to mitigate flood hazards. Normally, this type of flood studies does not consider regional effects on the local flood magnitudes and is based not only on outdated base flood events that miss new regional developments but also on localized flood control projects that overlook regional controls.

By representing a complete end-to-end integration that involves components from the initial meteorology and hydrology all the way to the subsequent hydraulic and flood delineation, it is possible to readily reflect the effects of basin wide developments or alterations on runoff generation. Therefore,

this integration provides a comprehensive framework for regional scale simulations that can potentially take into account regional effects on local flood studies.

Given the ability to generate outcomes at any specified modeling location, this development may also be used to support a more comprehensive generation of conventional flood damage assessment relationships like peak discharge frequency curves (discharge vs. exceedance probability) and rating curves (discharge vs. stage) at needed points on ungauged floodplains.

5.8 CONTRIBUTIONS TO TECHNOLOGY

This research provides important contributions to information technology and in particular to computing in civil engineering. The development of specific computational integration schemes with two external and full engineering simulation models that allow their inclusion as “tools” inside the ModelBuilder workflow model and framework. The computational ability to interface two standard engineering models has been implemented and adapted to comply with the needed workflow of processes to transform a precipitation map into a floodplain map. DLL applications in charge of executing the external calls have been developed based on the model’s interfacing functionalities. Therefore, a new computational scheme for the integration of independent modeling systems is demonstrated.

A library of functions for specialized modeling tasks, in the form of DLLs, was developed to accomplish and made available for code reuse in independent modeling exercises. In particular, utilities for the exchange of spatially-referenced

time series between NEXRAD, the HEC Data Storage System and the Arc Hydro data model (time series transfer routines), calling routines to two stand-alone and COM-compliant modeling systems (HEC-RAS and HEC-HMS callers), and geoprocessing models (an ensemble of standard GIS tools) to accomplish modular floodplain delineation sequences.

5.9 LIMITATIONS

The Map2Map computational framework that defines an automated workflow sequence for generating flood inundation maps is not an “assumption free” or a completely generic development. There are a various limitations and problems that may arise as the purpose for floodplain mapping changes based on the modeler’s objectives.

The computational scheme for the integration in Map2Map relies on two main aspects: The external execution of the two involved engineering models and the format structure of the ASCII text files defining the model’s input and output. As new versions of HEC-HMS and HEC-RAS become available, it might be possible to have changes in the way each model can be externally accessed and in the type of files and structure of the input and output files. The current development of Map2Map assumes a specific way to access HEC-HMS and HEC-RAS. In fact there is a fundamental distinction between the external execution of HMS and RAS as explained in sections 4.1.4 and 4.1.10. The external execution of RAS is done through a standard Shell function used to run executables by means of an application-specific DOS prompt command line instruction. New versions of HEC-HMS may change the syntax of the command

line needed for external execution and/or add new parameters to the DOS statement or even provide external access by means of an object library that exposes not only the execution options of the model but also additional functionalities like import and export operations. New HEC-RAS versions may also change the names and needed parameters of the properties and methods defined in the RAS object library (HEC-RAS controller) used to externally access the hydraulic model.

In addition to changes in access formats and structures, Map2Map can also be affected by changes in input and output file structures. Since on-the-fly file updates need to be done in Map2Map based on the characteristics of the new storm event being simulated, the content of the input and output files is modified accordingly. These file updates are based on current file structures that can be modified in new releases of both HMS and RAS.

As new versions of HMS and RAS are released, the aforementioned changes will need to be implemented in the Map2Map application to keep in tune with integrated external applications. The modular character of Map2Map provides a comprehensive framework that facilitates the computer code update needed to comply with new model's access mechanisms and file structures.

The current modeling integration for this research was based on steady-state hydraulic simulations that might be further extended to support unsteady state simulations for better tracking and description of the floodplain map dynamics as opposed to focusing only on the maximum extent of the flood inundation map. The steady state definition in Map2Map which utilizes the peak

value of the entire runoff hydrograph provides a conservative estimate of the flood maps that is only useful for worst case scenario type of analysis (for planning and design purposes) but it does not provide a realistic simulation of the floodplain as might be needed in warning and emergency response systems. The unsteady state definition however, provides a higher level of complexity since it only allows definition of a single routing reach at a time. By having a single reach definition, the unsteady state configurations will hamper the ability to define boundary conditions representing entering tributaries that supply additional runoff hydrographs to be superimposed and routed along the main channel system. To work around this unsteady restriction, several Map2Map definitions for each reach might be developed and nested to be sequentially executed by means of a script in charge of establishing the Map2Map batch process inside a loop for the entire number of river reaches. Even though viable, the aforementioned alternative to configure unsteady state simulations via nested Map2Map models for each reach may prove to be cumbersome and not robust under the current version of ArcGIS ModelBuilder.

The Map2Map simulation results are only as good as the modeling systems used by it. The intrinsic modeling limitations in HEC-RAS and HEC-HMS are inherited by Map2Map since it is only a framework for the integration of the modeling systems. Specific geomorphologic considerations that describe complex sedimentation processes in flooded over banks at river's delta areas can only be considered to the extent that the hydraulic simulation in HEC-RAS

allows. Thus, consideration of more complex river geomorphology issues may require the integration of a different and more objective-specific model.

5.10 FOLLOW-ON RESEARCH

Given the strong dependency of floodplain modeling and damage assessments on hydrologic engineering simulations, an integration scheme has been devised as a step forward in supporting additional components for flood impact analysis. Thus, besides the accomplished Hydrologic and Hydraulic modeling linkage, flood impact analysis components may now be implemented as an extension of the herein developed spatial framework. So the implementation of this spatially integrated hydrologic information system might be further expanded with flood information and impact analysis modules. This added functionality may assist in flood mitigation planning, and floodplain management by readily updating and evaluating new basin and floodplain scenarios.

NEXRAD datasets are being constantly updated and served in new formats over the internet (like the recently developed NEXRAD viewer and exporter for the NCDC radar archive). New radar precipitation formats need to be included as part of the Map2Map integration by developing computer code capable of ingesting the plethora of radar information publicly available.

An investigation on the potential scalable performance constraints as bigger watersheds or more detail input datasets are used for Map2Map implementations represents an interesting research topic. Scalable performance constraints due to watershed size and/or datasets' resolution may expose practical implementation limits on the time and space scales used to define a Map2Map

case study. Bigger watersheds (above the 75 square kilometers of Rosillo Creek) and/or newly available LIDAR datasets defining digital elevation models (DEMs) at finer resolutions over the over bank areas may impose important computer performance limitations. In a similar manner, a scalable level of effort in setting up Map2Map might be evaluated when implementing bigger watersheds in regional studies.

Appendix A

Tutorial on Map2Map Implementation

Prepared by Oscar Robayo
Center for Research in Water Resources
University of Texas at Austin
December, 2004

1 Introduction

This document is intended to be a step-by-step tutorial for the implementation of the Map2Map application for any given study area. It is expected that the user is familiar with ArcGIS software and knows how to perform basic operations such as adding data, opening attribute tables, selecting features, and editing the data. It is also expected that the user is familiar with the Arc Hydro data model for Water Resources (structural components, nomenclature, schema application, geometric and schematic network generations, building relationships, etc.) since the geographic integration of HMS and RAS is done through it.

2 Data Requirements

All the data necessary for this tutorial exercise are stored in data folder called Map2Map and available at:

<ftp://ftp.crwr.utexas.edu/pub/outgoing/Robayo/Map2Map> or in the accompanying CD as described in appendix B. The input datasets needed to execute this exercise are:

- The Map2Map pilot project dataset
- The precipitation event to be simulated
- An Arc Hydro data model for the study area
- The NEXRAD HRAP grid polygon dataset for the study area
- A preexisting HEC-HMS model setup for the study area
- A preexisting HEC-RAS model setup for the study area
- A digital elevation model (DEM) of the floodplain extent

3 Installing the Map2Map Pilot Project

The first step consists in installing the Map2Map pilot project on your local computer by following the steps below and then becoming familiar with it by looking at the data structure and executing the entire process.

The model builder toolbox (Map2Map.tbx) contains a model for going from a rainfall map to a flood inundation map. The processes involved are: Import of NEXRAD rainfall data into geodatabase with Arc Hydro data model, transfer data from NEXRAD cells to subbasins/catchments, export time series of precipitation to HEC-DSS, run the HEC-HMS model to obtain streamflow hydrographs at outlets. Import results from HEC-DSS into GIS, prepare and HEC-DSS file just for the HEC-RAS model, run the HEC-RAS model and generate a GIS compatible output file, import results back to GIS by means of an XML format, get water surface elevations from XML on cross sections, and then execute all the floodplain delineation routines to obtain the flood inundation polygon.

3.1 Copying the Map2Map directory

To install, first copy the entire Map2Map folder on the local hard drive under C:\. If you put the contents in another location, you might have to remap some of the tools used in the model. Even though relative pathnames were used to reference input files for each tool, sometimes the input locations STILL have to be remapped in your system.

3.2 Installing HEC-HMS and HEC-RAS

You should have installed the HEC-HMS and HEC-RAS modeling programs on your computer. These models are available for download from the HEC website at: <http://www.hec.usace.army.mil/default.html>.

3.3 Setting up the Modeling Directories

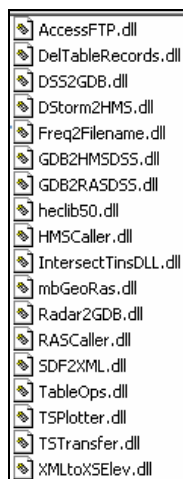
The installation of HEC-HMS will usually create a folder called hmsproj under C:\ (Unless the user defines a different project directory during installation). Copy the HEC-HMS project folder Salado_Unified from the HEC_Program_Installation folder included with Map2Map (C:\Map2Map\HEC_Program_Installation\hmsproj) into the hmsproj folder under C:\. So your HMS project directory will be located at C:\hmsproj\MyHMSproject

Copy the rasproj folder from the HEC_Program_Installation folder included with Map2Map to C:\. So your RAS project directory will be located at C:\rasproj\MyRASproject

After the models are installed and you have copied the pilot models in their respective locations make sure they run in stand-alone mode successfully.

3.4 Registering the DLLs

Register ALL of the DLLs in MapToMap\DLLs. Be aware that the third party DLL (heclib50.dll) is a non-registrable DLL that has to be placed under the “system32” directory location. One simple way to register DLLs is to run the **Register_in_menu.reg** file. Search for the **Register_in_menu.reg** file normally located in “C:\Program Files\ArcGIS\DeveloperKit\tools” and double click on it. By doing this, the mouse right click allows you to automatically register and unregister dlls for ArcGIS. The needed DLLs are shown below



3.5 Registering File Extensions in ArcGIS

In order to work in ModelBuilder you need to make all the needed file extensions known to ArcGIS so they can be referenced. In ArcCatalog, add the following file extensions that might be unknown to ModelBuilder: .dss, .sdf, and .prj (this one might be ambiguous as it is used as a HEC-RAS project file and also as a GIS projection file) file types by using the menus: Tool→Options→File Types.

3.6 Checking Accessibility and License Availability in ArcGIS

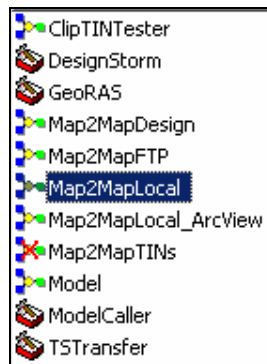
- Make sure HEC-HMS and HEC-RAS programs are NOT open
- Make sure the Spatial Analyst extension is turned on
- Make sure the 3D Analyst extension is turned on (for TIN manipulation)

3.7 Map2Map Usage

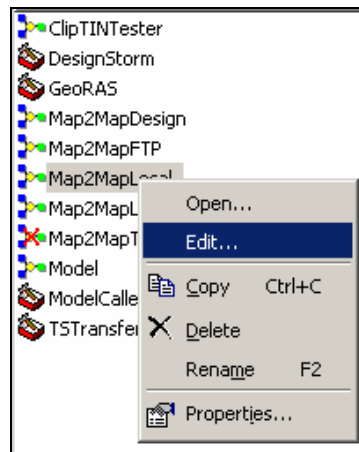
All of the model builder tools are in the MapToMap toolbox, located in the MapToMap folder. By using ArcCatalog you can access this toolbox which in turn contains the following toolsets:

- TSTransfer - contains tools for transferring time series data from one format or feature class to another
- ModelCaller - contains tools for calling HEC-HMS and HEC-RAS.
- GeoRAS - contains tools which mimic the functionality provided by the GeoRAS toolbar for ArcMap related to GIS post-processing of hydraulic input to generate the flood inundation map.
- Map To Map model – Main model for converting rainfall data to a flood inundation polygon.

In ArcCatalog look for the Map2Map Model you want to execute from the current available versions, that is: Map2MapLocal (running from precipitation input locally stored), Map2MapFTP (running from precipitation input located at remote FTP server), Map2Map_ArcView (same as Map2MapLocal but for those with just ArcView license, and Map2MapDesign (running from statistical design storms for Texas).



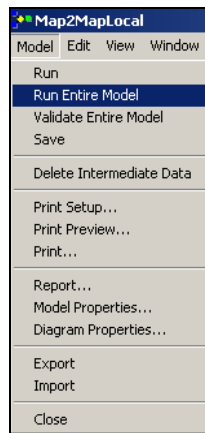
In ArcCatalog right click on the Model you want to execute to open the model's workflow for it.



The Map2Map workflow for the historic event use case is shown below:



Once the Model is open click on the “Model” menu and select “Run Entire Model” as shown below to execute all the components (rectangle processes).



Congratulations! You have successfully implemented Map2Map for the pilot project.

4 Update the Precipitation Event

If you are not going to use the same storm event used for the pilot (at beginning of July, 2002) then you must copy the new files with the same formatting structure under the “C:\Map2Map\RawNEXRAD_Data\NEXRAD_data” folder location. Next you need to generate a new Date File List (DateFListPart.txt) having the time window of your new storm event and place it in the “C:\Map2Map\RawNEXRAD_Data” folder location.

5. Generate an Arc Hydro Data Model for the Study Area

This exercise requires some basic knowledge of the Arc Hydro data model for water resources. So in principle the user must develop a complete Arc Hydro for the study area. However, some “shortcuts” can be implemented to develop a smaller version of Arc Hydro with just the main components being used in the Map2Map integration (the author recommends a complete Arc Hydro development once the user is familiar with all the components).

It is necessary to have in Arc Hydro the Watershed (under Drainage feature dataset), HydroEdge, and HydroJunction (under Network feature dataset) feature classes that resemble the topology used in HEC-HMS.

It is also necessary to have in Arc Hydro the CrossSection feature class (under Channel feature dataset) that resembles the cross sections defined in HEC-RAS to describe the river morphology.

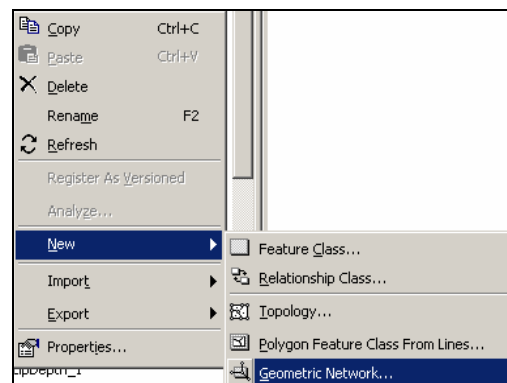
The HydroEdge feature class must have a flow direction assigned to it by using standard ArcHydro tools to assign and store the flow directions.

5.1 Applying the Schema (Arc Hydro Framework)

Following standard Arc Hydro procedures the user is encourage to apply the simplified schema to the created geodatabase to provide it with necessary fields to be later populated like HydroIDs and JuntionsIDs.

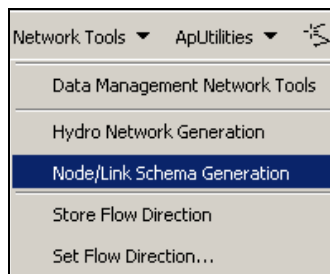
5.2 Generating the Geometric Network

The user need to create an Arc Hydro Geometric Network based on the previous Hydro Edge and Hydro Junction feature classes that match the HEC-HMS topology. ArcCatalog is used for creating the geometric network by right clicking on the feature dataset containing the involved feature classes (Network feature dataset) and selecting the New > Geometric Network options as shown below and then following the wizard.



5.3 Generating the SchematicNetwork

After the geometric network has been created a schematic version of it needs to be generated to match the hydrologic setup in HEC-HMS. Under the Network Tools menu in the ArcHydro toolbar you may generate a SchematicNetwork by using the “Node/Link Schema Generation” option in the Arc Hydro toolbar.



Once the simplified version of Arc Hydro (the Framework) has been developed for the study area the user needs to place it at the right Map2Map location for later reference. The Map2Map folder location is: “C:\Map2Map\MyArcHydro.mdb.”

5.4 Populating HydroID and JunctionID fields

Using the Arc Hydro toolbar the user must assign HydroIDs to HydroEdge, HydroJunction, and Watershed feature classes as well as JunctionIDs to the Watershed feature class.

5.5 Populating NextDownID field

Using the Arc Hydro toolbar (Attribute Tools > Find Next Downstream Junction) the user must assign NextDownIDs to the HydroJunction feature class and double check that the geographic topology matches the hydrologic topology in HMS.

5.6 Populating the SchematicNode Feature with HMSCodes

In ArcMap open the attribute table of the SchematicNode feature class and add a text field called “HMSCode.” Populate the field HMSCode for those nodes representing flow change points (hydrograph entry points) with the exact modeling name given in HEC-HMS which also corresponds to the B part in HEC-DSS pathnames.

5.7 Populating the Watershed Feature with HMSCodes

In ArcMap open the attribute table of the Watershed feature class and add a text field called “HMSCode.” Populate the field HMSCode for all the subbasins with the exact modeling name given in HEC-HMS. This way Map2Map will generate a meteorologic file with hyetographs for each subbasin.

5.8 Populating the CrossSection Feature with RASCodes

In ArcMap open the attribute table of the CrossSection feature class and add a field called “RASCode.” Populate the field RASCode, for those cross section representing flow change points (hydrograph entry points) with the complete modeling identifier name given in HEC-RAS and corresponding to the concatenation of the StreamID, Reach ID, and Station identifiers separated by a comma.

5.9 Building the HydroJunctionHasWatershed Relationship

The JunctionID field in the Watershed feature class must be populated with the HydroID of the HydroJunction representing its outlet. By means of this field association the relationship class “HydroJunctionHasWatershed” can be built. To build the relationship class the user may use standard functionality in ArcCatalog.

5.10 Building the HydroJunctionHasSchematicNode Relationship

The FeatureID field in the SchematicNode feature class is populated with the HydroID of the HydroJunction it represents. By means of this field association the relationship class “HydroJunctionHasSchematicNode” can be built. To build the relationship class the user may use standard functionality in ArcCatalog.

5.11 Building the HydroJunctionHasCrossSection Relationship

The JunctionID field in the Cross Section feature class needs to be populated with the HydroID of the HydroJunction that is exchanging flow into a given downstream cross section. By means of this field association the relationship class “HydroJunctionHasCrossSection” can be built. To build the relationship class the user may use standard functionality in ArcCatalog.

5.12 Copying the Time Series Tables

There are two time series tables in Arc Hydro, TimeSeries and TSType tables (created by the schema for you) that need to be in the Arc Hydro geodatabase for your project. In addition to these two standard tables you need to have and additional table to store the HEC-DSS descriptors called the DSSTSCatalog (or DSSTSType) so copy this table from the pilot Arc Hydro (located at C:\Map2Map\RosilloFull.mdb) into your Arc Hydro geodatabase.

6 Get HRAP grid Polyline layer of the Study Area

Using the HRAP layer for Texas (located at C:\Map2Map\RawNEXRAD_Data\NEXRADLayerTX) select only the NEXRAD

cells that cover your study area and place it inside your ArcHydro geodatabase at “C:\Map2Map\RosilloFull.mdb\Hydrography\HydroResponseUnit.”

7 Update the PixelID file list

Based on the NEXRAD cells that overlay your study area you need to update the pixelID (six digit number) file list (PixelIDs.txt) located at C:\Map2Map\RawNEXRAD_Data. Make sure the PixelIDs for your study area actually have rainfall data for them (not dry cells) in the hourly files located at “C:\Map2Map\RawNEXRAD_Data\NEXRAD_data” location.

8 Generate Intersection between HRAP and Watershed Polygons

For the “Time Series Transfer” script tool to work, a polygon layer that contains the intersection of the NEXRAD cells and the Watersheds of the study area is needed. Standard GIS functionality can be used to generate this layer (Analysis Tools > Overlay > Intersect hammer) which is called “IntRadWsh” in the pilot. Once it is generated place it in the Arc Hydro geodatabase at: “C:\Map2Map\RosilloFull.mdb\Hydrography.” Now you need to add four fields to your IntRadWsh layer: KEYFROM, KEYTO, PCTFROM, and PCTTO the first two as long integers and the last two as double type.

9 Add HEC-HMS Model Setup for the Study Area

Add the HEC-HMS project setup of your study area to the default directory location containing all the hydrologic HMS projects (C:\hmsproj). Your project will be placed at “C:\hmsproj\MyHMSProject.” Make sure the project is running properly when executed from the HMS user interface in stand-alone mode.

10 Add HEC-RAS Model Setup for the Study Area

Add the HEC-RAS project setup of your study area to the default directory location containing all the hydraulic RAS projects (C:\rasproj). Your project will be placed at “C:\rasproj\MyRASProject”. Make sure the project is running properly when executed from the RAS user interface in stand-alone mode.

11 Add the Digital Elevation Model (DEM) of the Floodplain Area

The continuous surface representing the terrain elevations of your floodplain area is a vital piece of input for the final floodplain delineation phase. In order to obtain reasonable results, the resolution of the terrain raster must be high enough to appropriately define places underwater. Regularly, the DEM must have a

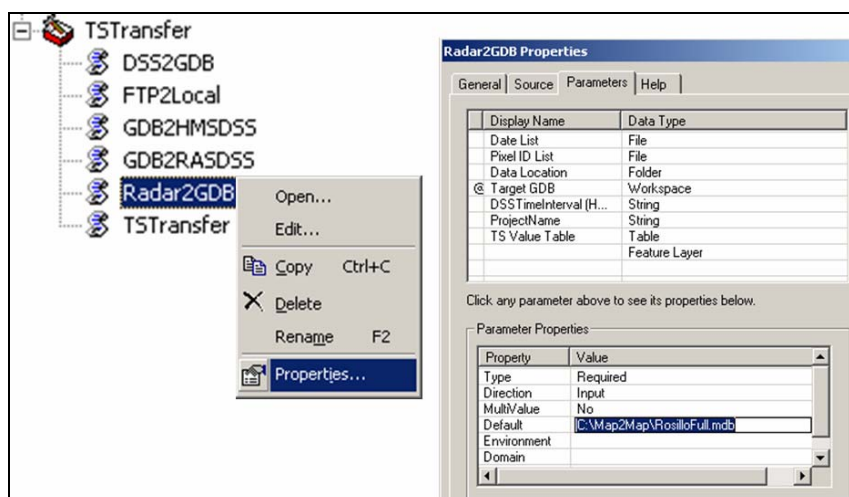
resolution consistent with the level of resolution used to define the HEC-RAS cross sections. This resolution is usually a 20 feet square grid resolution or less.

The selected raster DEM must be available to Map2Map by placing it at the “C:\Map2Map\Grids_TINs” directory location. Make sure the DEM has the same projection used in the Arc Hydro geodatabase.

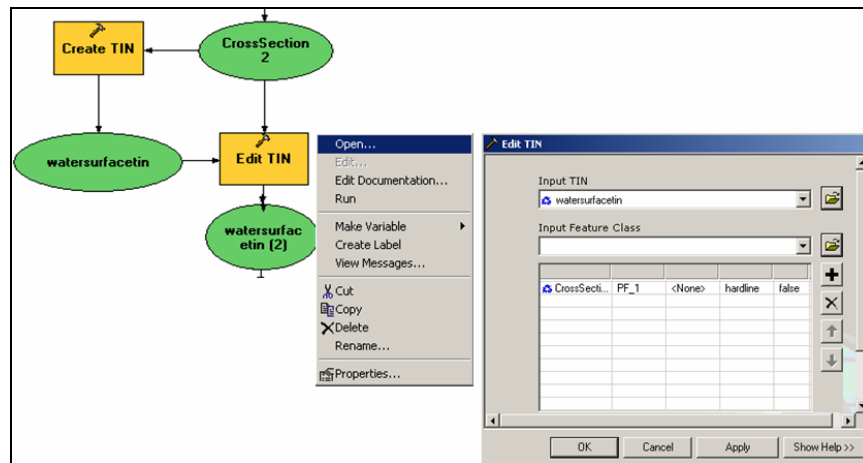
12 Update the Dataset Locations in ModelBuilder

Now that you have made available to Map2Map all the datasets pertaining to your very own study area it is time to change all the input source locations and parameters to reference the data and parameters of your own project.

To change the input source locations and parameters in Map2Map the user can use ArcCatalog or ArcMap to open the Map2Map Model as described in section 3.7. Once the Map2Map model is open the user can manually change all the inputs (datasets and parameters). This task needs to be done in two ways. First the user needs to modify the default parameter setup of all the script tools (user-defined tools) as shown in the Figure below. In this example the user has selected the script tool “Radar2GDB” and selected its properties from the right click menu. Once the Properties window is open, the user needs to select each parameter name and change the default value of each property as shown below for the Target GDB property name. That is the user needs to make sure the properties reflect the new input dataset that wants to be implemented. For this example the user must change “C:\Map2Map\RosilloFull.mdb” by “C:\Map2Map\MyBasin.mdb.”

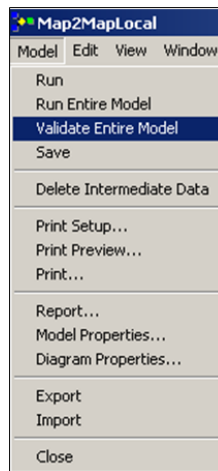


Second, the user needs to modify the default parameter setup of all the standard tools (GIS hammer tools) as shown in the Figure below by right clicking on the “yellow rectangle”, selecting “Open” and then modifying the name and location of the input datasets (remapping your new input locations) or assigning the value of the new input parameters. An example is shown below for the “Edit TIN” standard tool which has been open and its parameters have been change to reflect the new basin information and locations.



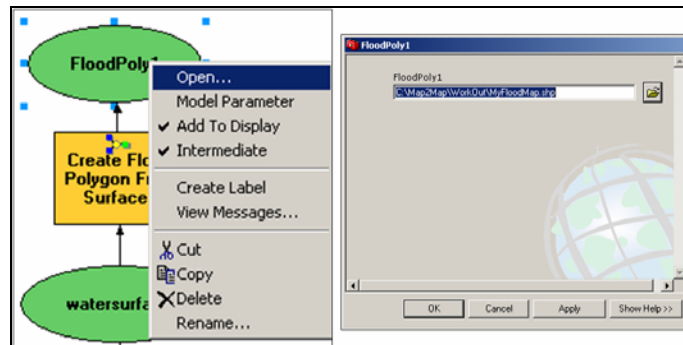
It is a good idea and practice to double check all the oval’s inputs including not only the blue ovals but also the internally generated green ovals to make sure they conform to your project.

After all the input has been remapped the user may open the Map2Map Model and select “Model” and then select “Validate Entire Model” to insure the model implements all the new inputs of the new basin. This is shown in the Figure below.



13 Execute Map2Map for New Study Area

Using ArcCatalog or ArcMap execute Map2Map for your specific study area by following the process described in section 3.7. The user may change the name of the output floodplain map by right clicking on the final green oval representing the output flood map and changing its name as shown in the figure below. If you use ArcMap to execute Map2Map make sure the Arc Hydro dataset being referenced corresponds to your geodatabase inside the Map2Map directory location.



Congratulations! You have successfully implemented Map2Map for your very own study area and modeling purposes.

Appendix B

Map2Map CD Content

1 CD Data Structure

1.1 Main Files

The main files stored in the Map2Map CD are:

- **Readme.doc** – contains a step-by-step tutorial for implementing Map2Map on the Rosillo Creek pilot project
- **Map2Map.tbx** – The Map2Map toolbox for ArcGIS 9 Model Builder
- **RosilloFull.mdb** – Geodatabase with Arc Hydro data model for the Rosillo Creek pilot case study
- **Map2Map.mxd** – ArcMap document with saved configuration for Rosillo Creek from which Map2Map can be executed

1.2 Data Directories

The Map2Map CD contains the following data directories:

- **DLLs** – folder that contains all DLLs used by the customized script tools
 - Radar2GDB.dll
 - TSTransfer.dll
 - GDB2HMSDSS.dll
 - HMSCaller.dll
 - DSS2GDB.dll
 - GDB2RASDSS.dll
 - RASCaller.dll
 - SDF2XML.dll
 - XML2XSElev.dll
 - Freq2Filename.dll
 - TableOps.dll
 - DelTableRecords.dll
 - DStorm2HMS.dll
 - Heclib50.dll
 - AccessFTP.dll

- **ScriptFiles** – folder with all script files associated with DLL components
 - Radar2GDB.vbs
 - TSTransfer.vbs
 - GDB2HMSDSS.vbs
 - HMSCaller.vbs
 - DSS2GDB.vbs
 - GDB2RASDSS.vbs
 - RASCaller.vbs
 - SDF2XML.vbs
 - XML2XSElev.vbs
 - Freq2Filename.vbs
 - GetFreqSeries.vbs
 - AppendTable.vbs
 - DeleteTableRecords.vbs
 - DStorm2HMS.vbs
 - CallDStorm.vbs
 - AccessFTP.vbs
- **SourceCode** – folder that contains the source code for the user-defined DLLs
 - Radar2GDB.vbp
 - TSTransfer.vbp
 - GDB2HMSDSS.vbp
 - HMSCaller.vbp
 - DSS2GDB.vbp
 - GDB2RASDSS.vbp
 - RASCaller.vbp
 - SDF2XML.vbp
 - XML2XSElev.vbp
 - Freq2Filename.vbp
 - TableOps.vbp
 - DelTableRecords.vbp
 - DStorm2HMS.vbp
 - AccessFTP.vbp
- **RawNEXRAD_Data**: Contains the NEXRAD precipitation input text files for the July 2002 storm event in Texas
- **Grids_TINs**: Contains rasters and TINs needed to support the computations in Map2Map
- **WorkOut**: Contains intermediate files like rasters, and TINs to support Map2Map computations
- **HEC_Program_Installation** - contains setup programs for installing HEC-HMS and HEC-RAS

- hmsproj - contains folders with HEC-HMS setup files for diverse user-defined and sample basins like the Salado Creek pilot project
 - rasproj – contains folders with HEC-RAS setup files for a diverse user-defined and sample river segments like the Rosillo Creek pilot project
- **DigiMaps** – Contains design storms for Texas in the form of grid rainfall maps generated by Halff Associates, Inc.
- **Docs_Shows** – Documents and Power Point presentations describing the Map2Map application

2 ModelBuilder Models

Inside the Map2Map toolbox (Map2Map.tbx) there are various ModelBuilder Models (workflow of sequential processes) representing several Map2Map versions. A brief description of these model versions follows:

Map2MapLocal: Map2Map Model application that is being fed with precipitation data stored in the local computer system.

Map2MapFTP: Map2Map Model application that is being fed with precipitation data stored in a remote FTP server.

Map2MapDesign: Map2Map Model application that is being fed with synthetic precipitation data in the form of grid rainfall maps for various frequencies and durations.

Map2MapLocal_ArcView: Map2Map Model application that is being fed with precipitation data stored in the local computer system and that uses only standard GIS tools available under Arc View license.

References

- Ackerman, Cameron, T., Evans, Thomas, A., and Brunner, Gary, W. 1999. HEC-GeoRAS: Linking GIS to Hydraulic Analysis Using ARC/INFO and HEC-RAS. 1999 International ESRI User Conference. ESRI, Redlands, CA.
<<http://gis.esri.com/library/userconf/proc99/proceed/papers/pap682/p682.htm>>
- Ackerman, Cameron. 2004. Flood Warning and Response System for the Susquehanna River. AWRA Spring Specialty Conference, May 17-19, 2004, Nashville, Tennessee.
- Anderson, David, J. 2000. GIS based hydrologic and hydraulic modeling for floodplain delineation at highway river crossings. M.S. thesis. The University of Texas at Austin. Center for Research in Water Resources online report 00-11. <<http://www.crwr.utexas.edu/reports/2000/rpt00-11.shtml>>
- Arctur, David and Zeiler Michael. 2004. Designing Geodatabases: Case Studies in GIS Data Modeling. ESRI Press, Redlands, California.
- Azagra, Esteban, Olivera, Francisco, and Maidment, David. 1999. Floodplain Visualization using TINs. Center for Research in Water Resources online report 99-5. <<http://www.crwr.utexas.edu/reports/1999/rpt99-5.shtml>>
- Beard, L. R. 1997. Estimating Flood Frequency and Average Annual Damage. Journal of Water Resources Planning and Management, Volume 123, No. 2, pp. 84-88.
- Bedient, P. B., and Huber, W. C. 1992. Hydrology and Floodplain Analysis. 2nd edn, Wokingham, UK, Addison-Wesley.
- Boyle, S. J., Tsanis, I. K., and Kanaroglou, P. S. 1998. Developing Geographic Information Systems for Land Use Impact Assessment in Flooding Conditions. Journal of Water Resources Planning and Management, Volume 124, No. 2, pp. 89-98.
- Brimicombe, A. J., and Bartlett, J. M. 1996. Linking GIS with Hydraulic Modeling for Flood Risk Assessment: The Hong Kong Approach. In GIS and Environmental Modeling: Progress and Research Issues, edited by

- Goodchild, M.F., Steyaert, L.T., and Parks, B.O., Fort Collins, CO, GIS World Books, pp. 165-168.
- Danish Hydraulic Institute (DHI). 2000. MIKE 11 Reference Manual and User Guide, 2000.
- Danish Hydraulic Institute (DHI). 2001. MIKE 21 Reference Manual and User Guide, 2001.
- Davis, S. A., Johnson, N. B., Hansen, W. J., Warren, A., Reynolds, F. R., Foley, C. O., and Fulton, R. L. 1988. National Economic Development Procedures Manual: Urban Flood Damage (Alexandria, VA, U.S. Army Corps of Engineers, Institute for Water Resources).
- Djokic, D., Beavers, M., A., and Deshakulakarni, K. 1994. Arc/HEC2- An ArcInfo-Hec2 Interface. Proceedings of the 21st annual conference on water policy and management. American Water Resources Association, ASCE, pp. 41-44, May 1994.
- Djokic, D., Coates, A., and Ball, J. E. 1995. GIS as integration tool for hydrologic modeling: A need for generic hydrologic data exchange format. 1995 ESRI User Conference. Redlands, CA, pp. 22-26, May 1995.
- Fread, D., L., and Lewis, J., M. 1988. FLDWAV: A Generalized Flood Routing Model. Proceedings of National Conference on Hydraulic Engineering. Colorado Springs, Colorado.
<http://www.nws.noaa.gov/oh/hrl/rvrmech/papers/hl_261.pdf>
- GRASS 4.1 User's Reference Manual. 1993. U.S. Army Corps of Engineers Construction Engineering Research Laboratory, Champaign, Ill.
- Grigg, N. S., and Helweg, O. J. 1975. State-of-the-Art of estimating flood damage in urban areas. Water Resources Bulletin. Volume 11, No. 2, pp. 379-390.
- Haan, C. T. 1977. Statistical Methods in Hydrology, Iowa State University Press, Ames, Iowa.
- Hansen, W. J. 1987. National Economic Development Procedures Manual: Agricultural Flood Damage (Alexandria, VA, U.S. Army Corps of Engineers, Institute for Water Resources).
- Harold, E. R., and Means, W. S. 2001. XML in a Nutshell, O'Reilly & Associates, Inc., Sebastopol, CA.

- Heinzer, Tom, Sehbat, Michael, Feinberg, Bruce, and Kerper, Dale. 2000. The Use of GIS to Manage LIDAR Elevation Data and Facilitate Integration with the MIKE21 2-D Hydraulic Model in a Flood Inundation Decision Support System. Proceedings of the 2000 ESRI users' international conference. San Diego, CA, June 2000.
<<http://gis.esri.com/library/userconf/proc00/professional/papers/PAP675/p675.htm>>
- Huang B., and Jiang B. 2002. AVTOP: a Full Integration of TOPMODEL into GIS. *Environmental Modelling and Software*, Volume 17, pp. 261-268.
- Jonge, Tineke De (Delft Hydraulics); Kok, Matthijs; and Hogeweg, Marten. 1996. Modelling floods and damage assessment using GIS. *Application of Geographic Information Systems in Hydrology and Water Resources Management*, IAHS Publication (International Association of Hydrological Sciences), Volume 235, pp. 299-306.
- Maidment, D. R. 1993. GIS and Hydrologic Modeling. In *Environmental Modeling with GIS*, edited by Goodchild M.F., Parks B.O., and Steyaert L.T., New York, Oxford University Press, pp. 147-167.
- Maidment, D. R. 2002. *Arc Hydro-GIS for water resources*. ESRI Press, Redlands, California.
- Muller, H. G., and Rungoe, M. 1995. Integrating Floodplain Management and Numerical Modeling using ArcView. Proc., 15th Annu. ESRI User Conf., Environmental Systems Research Institute, Redlands, California.
- Nelson, E. J. 2000. WMS v6.0 Reference Manual. *Environmental Modeling Research Laboratory*, Brigham Young University, Provo, Utah, pp. 487.
- Noman N. S., Nelson E. J., and Zundel A. K. 2001. Review of Automated Floodplain Delineation from Digital Terrain Models. *Journal of Water Resources Planning and Management.*, Volume 127, No. 6, pp. 394-402.
- Obenour D., Maidment, D., Evans, T., Yates D., 2004. An Interface Data Model for HEC-HMS. AWRA Spring Specialty Conference, May 17-19, 2004, Nashville, Tennessee.
- Olivera, F., and Maidment, D. R. 2000. GIS Tools for HMS modeling support. Chapter 5, *Hydrologic and hydraulic modeling support with geographic information systems*, D.R. Maidment and D. Djokic, eds., ESRI Press, San Diego, Calif.

- Olsen R. J., Beling, P. A., Lambert, J. H., and Haimes, Y. Y. 1998. Input-Output Economic Evaluation of System of Levees. *Journal of Water Resources Planning and Management*, Volume 124, No. 5, pp. 237-245.
- Penning-Rowsell, E. C., and Chatterton, J. B. 1977. *The Benefits of Flood Alleviation: A Manual of Assessment Techniques*, Gower Publishing Company Limited, Aldershot, England.
- Raman V., Crone C., Haas L., Malaika S., Mukai T., Wolfson D., and Baru C. 2002. Data Access and Management Services on Grid. <http://www.cs.man.ac.uk/grid-db/papers/dams.pdf>, pp. 1-24.
- Reed, S., Maidment, D. R. 1999. Coordinate Transformations for Using NEXRAD Data in GIS-based Hydrologic Modeling. *Journal of Hydrologic Engineering*, Volume 4, No. 2, pp. 174-182.
- Reitsma, R. F., Sautins, A. M., and Wehrend, S. C. 1994. Construction kit for visual programming of river-basin models. *Journal of Computing in Civil Engineering*, ASCE, Volume 8, No. 3, pp. 378-384.
- Runge, Morten, and Olesen, Kim Wium. 2003. Combined one- and two-dimensional flood modeling. Fourth Iranian Hydraulic Conference, 21-23 October, Shiraz, Iran.
- Sinha S. K., Sotiropoulos F., and Odgaard A. J. 1998. Three-dimensional numerical model for flow through natural rivers. *Journal of Hydraulic Engineering*. Volume 124, No. 1, pp. 13-24.
- Snead, Daniel, B. 2000. Development and application of unsteady flood models using geographic information systems. Departmental Report. The University of Texas at Austin. Center for Research in Water Resources online report 00-12. <<http://www.crwr.utexas.edu/reports/2000/rpt00-12.shtml>>
- Southwest Florida Water Management District's Watershed Management Program Guidelines and Specifications. 2002.
- Sui, D. Z., and Maggio, R. C. 1999. Integrating GIS with hydrological modeling: practices, problems, and prospects. *Computers, Environment and Urban Systems*, Volume 23, pp. 33-51.

- Tate, E. C., Maidment, D. R., Olivera, F., and Anderson D. J. 2002. Creating a terrain model for floodplain mapping. *Journal of Hydrologic Engineering*. Volume 7, No 2, pp. 100-108.
- Tisdale, T. S. 1996. Object-Oriented Analysis of South Florida Hydrologic Systems. *Journal of Computing in Civil Engineering*, ASCE, Volume 10, No. 4, pp. 318-326.
- U. S. Army Corps of Engineers (USACE). 1988. Flood Damage Analysis Package, User's Manual. Hydrologic Engineering Center, Davis, CA.
- U. S. Army Corps of Engineers (USACE), 1991. HECLIB, Volume 2: HEC-DSS subroutines, Programmer's Manual, CPD-57, Hydrologic Engineering Center (HEC), Davis, CA.
- U. S. Army Corps of Engineers (USACE), 1995. HEC-DSS, User's Guide and Utility Manuals, User's Manual, CPD-45, Hydrologic Engineering Center, Davis, CA.
- U. S. Army Corps of Engineers (USACE). 1997. HEC-RAS River Analysis System, User's Manual. Hydrologic Engineering Center, Davis, CA.
- U. S. Army Corps of Engineers (USACE). 1998. Flood Damage Analysis Package, User's Manual. Hydrologic Engineering Center, Davis, CA.
- U. S. Army Corps of Engineers (USACE). 2000a. HEC-HMS Hydrologic Modeling System, User's Manual. Hydrologic Engineering Center, Davis, CA.
- U. S. Army Corps of Engineers (USACE). 2000b. Geospatial Hydrologic Modeling Extension HEC-GeoHMS, User's Manual, CPD-77. Hydrologic Engineering Center, Davis, CA.
- U. S. Army Corps of Engineers (USACE). 2002a. HEC-GeoRAS: An Extension for Support of HEC-RAS using ArcView. Users Manual. Version 3.1, CPD-76. October 2002, Hydrologic Engineering Center, Davis, California.
- U. S. Army Corps of Engineers (USACE). 2002b. HEC-RAS: River Analysis System. Hydraulic Reference Manual, Version 3.1, November, Hydrologic Engineering Center, Davis, California.

- Werner, M. G. F. 2001. Impact of grid size in GIS based flood extent mapping using a 1D flow model. *Physics and Chemistry of the Earth, Part B: Hydrology, Oceans and Atmosphere*. Volume 26, Issues 7-8, pp. 517-522.
- Whiteaker, T. L., and Maidment, D. R. 2001. A Prototype Toolset for the ArcGIS Hydro Data Model. Center for Research in Water Resources, Online Publication, CRWR Online Report 01-6, pp. 170.
- Whiteaker, T. 2003. Hydrologic Simulation with Arc Hydro, libhydro, and Model Builder. Center for Research in Water Resources, University of Texas at Austin,
<http://www.crwr.utexas.edu/gis/gishydro03/LibHydro/HydrologicSimulationWithArcHydrolibHydroandModelBuilder.doc>.
- Wurbs, R. A. 1996. Optimal Sizing of Flood Damage Reduction Measures Based on Economic Efficiency. *Water Resources Development*, Volume 12, No. 1, pp. 5-16.
- Wurbs, R. A., Toneatti S., and Sherwin J. 2001. Modeling Uncertainty in Flood Studies. *Water Resources Development*, Volume 17, No. 3, pp. 353-363.
- Yang, C. R., and Tsai C. T. 2000. Development of a GIS-Based Flood Information System for Floodplain Modeling and Damage Calculation. *Journal of the American Water Resources Association*, Volume. 36, No. 3, pp. 567-577.
- Zeiler, Michael. 1999. *Modeling Our World: The ESRI Guide to Geodatabase Design*. ESRI Press, Redlands, California.

Vita

Oscar Robayo was born in Cartago, Colombia on June 24, 1966, the son of Oscar Robayo and Alicia Amado. After completing his work at Calasanz High School, Cucuta, Colombia, in 1984, he entered Universidad Francisco de Paula Santander in Cucuta, Colombia. He received the degree of Bachelor of Science in Civil Engineering from Universidad Francisco de Paula Santander in May, 1991. Upon graduation, he went to the Colombian capitol to pursue a post-graduate degree in Civil Engineering from Universidad de los Andes, Bogota, Colombia. In March, 1994, he graduated with Magister in Civil Engineering from The Universidad de los Andes in March, 1994. In September, 1996 he enrolled in the University of California at Davis for graduate studies in Hydrologic Sciences and graduated with a Master of Science in Hydrologic Sciences in December, 1997. He entered the Graduate School of The University of Texas at Austin in August, 2001.

Permanent address: 3471 Lake Austin Blvd, Apt. B,
Austin, TX 78703

This dissertation was typed by the author.